Eventus®

Version 6.3C

Software for Event Studies

and CRSP Data Retrieval

http://www.eventstudy.com/

User's Guide

June 2000 Edition

Revised Printing, October 2000

copyright © 1989–2000 Cowan Research, L.C.

Notice Concerning Use of **Eventus** Software

Eventus® is a proprietary software product, © 1989–2000 by Cowan Research, L.C. Eventus software is licensed and not sold. Eventus software is licensed to an organization or an individual to be used only by the licensee, or in the case of an organization, its employees, faculty and students as applicable. The software is not to be copied (except in such manner as may be expressly permitted by the license), sold and/or given to anyone. Eventus is a registered trademark for software sold by Cowan Research, L.C. Registered refers to U.S. trademark registration; Australian trademark registration applied for. EventStream, request file, and `USERSTOK` are trademarks of Cowan Research, L.C.

Eventus licensees may reproduce this manual for internal use only, provided that each copy contains this copyright notice page.

ISBN 1–893112–09–8

Contents

1 Introduction **1**

2 Event Studies: The Essentials **3**

3 Event Studies: The Options **13**
3.1 Event Studies Centered on a Single Event Date 13
3.2 An Event Study Example 31
3.3 Abnormal Returns between Paired Events: The TWIN Option . 37
3.4 Reprinting or Merging Saved Event Studies with the OLDSTUDY Statement 40

4 Event Studies Using Non-CRSP Data **43**
4.1 Event Studies Centered on a Single Event Date 43
4.2 An Event Study Example 54
4.3 Abnormal Returns between Paired Events: The TWIN Option . 59

5 Extracting Event Study Results for Further Analysis **63**
5.1 The EVENTUS Statement 63
5.2 The WINDOWS Statement 64
5.3 The EXTRACT Statement 64
5.4 Usage example 67

6 Event Studies Using the Event Parameter Approach **71**
6.1 Statements for the Event Parameter Approach 71
6.2 An Event Parameter Approach Example 78

7 Obtaining Returns, Prices, Volume, Number of Trades and Shares Outstanding from the CRSP Database **83**
7.1 The EVENTUS statement 83

7.2 The REQUEST statement 84
7.3 The RETURNS, PRICES and VOLUME statements 87

8 Converting Calendar Dates to CRSP Trading Day or Month Numbers Using DATECONV 93
8.1 The EVENTUS statement 93
8.2 The DATECONV statement 94

9 Converting CUSIP Identifiers Using CUSIPERM 99
9.1 The EVENTUS statement 100
9.2 The CUSIPERM statement 100

A Technical Reference 103
A.1 Event Study Prediction Errors 103
A.2 Event Study Test Statistics 105
A.3 Different estimation and event return intervals 115
A.4 Variable Names In Eventus Output SAS Data Sets 116
A.5 Missing Returns 118

B Reference Guide to Eventus Statements 121
B.1 CUSIPERM Statement 121
B.2 DATECONV Statement 122
B.3 EVENTUS statement 126
B.4 EVTSTUDY Statement 130
B.5 EXTRACT Statement 136
B.6 OLDSTUDY Statement 138
B.7 PRICES Statement 141
B.8 REQUEST Statement 145
B.9 RETURNS Statement 152
B.10 TITLE and TITLE2 Statements 154
B.11 UPCUSIP Statement 154
B.12 VOLUME Statement 155
B.13 WINDOWS Statement 158

C How Eventus Finds the CRSP Stock Database 159
C.1 CRSPAccess 159
C.2 SFA Format 160

C.3 Size index files used with both CRSPAccess and SFA format databases 160
C.4 CRSP data permanently stored in SAS data sets 161

List of Tables

A.1 Adjustments factors for different estimation and event return intervals. 116
A.2 Variable names in a saved event study data set. 117
A.3 Eventus special SAS missing values for missing returns from the CRSP database. 119

B.1 PACKAGE specifiers for the EVTSTUDY statement. 134

List of Figures

2.1 The simplest Eventus event study program. 4
2.2 Request file: PERMNOs and dates, illustrating tolerated irregularities. 4
2.3 Sample Eventus output: First page. 6
2.4 Sample Eventus output: Second page. 7
2.5 Sample Eventus output: Third page. 8
2.6 Sample Eventus output: Fourth page. 9
2.7 Sample Eventus output: Fifth page. 10
2.8 Sample Eventus output: Final page. 11

3.1 Eventus statements for an event study centered around a single date. 14
3.2 Eventus program for an event study of the "Too Big to Fail" policy. 32
3.3 Request file for the "Too Big to Fail" study. 33
3.4 The first page of Eventus output. 34
3.5 The Eventus sample listing and input report. 35
3.6 The Eventus parameter estimate listing. 36
3.7 Event study results. 38
3.8 Eventus statements for event studies cumulating returns between two firm-specific event dates. 39
3.9 Eventus statements for reprinting and merging saved event studies. 40
3.10 Sample Eventus program using OLDSTUDY to merge two saved event studies. 42

4.1 Eventus statements for an event study centered around a single date. 44

4.2 Eventus program for an event study of the "Too Big to Fail" policy. . . . 55
4.3 Part of the request file for the "Too Big to Fail" study. . . . 56
4.4 First 62 lines of the user-supplied returns file for the "Too Big to Fail" study. . . . 57
4.5 The Eventus sample listing and input report. . . . 58
4.6 The Eventus parameter estimate listing. . . . 59
4.7 Event study results. . . . 60
4.8 Eventus statements for event studies cumulating returns between paired event dates. . . . 61

5.1 Eventus statements for extracting saved event study results. . . . 64
5.2 Example using EXTRACT to organize firm-by-firm event study results for further analysis. . . . 68
5.3 Contents of SAS data set abnormalreturns produced by Figure 5.2 code. . . . 69

6.1 Eventus statements for an event parameter approach event study. 72
6.2 Eventus program to assemble data for the event parameter example. . . . 77
6.3 Request file for the event parameter approach demonstration. 78
6.4 Eventus program to complete the event parameter example. . . . 79
6.5 First part of the results for the event parameter approach demonstration. . . . 80
6.6 Remaining results for the event parameter approach demonstration. . . . 81

7.1 Eventus statements to read stock returns from a CRSP database. 84
7.2 Eventus statements to read stock prices from the CRSP database. 85
7.3 Eventus statements to read trading volume data from the CRSP database. . . . 86

8.1 Eventus statements for converting calendar dates to CRSP trading day or month numbers. . . . 94

9.1 Eventus statements to convert CUSIP identifiers to CRSP permanent identification numbers. . . . 100

Chapter 1

Introduction

Eventus® performs event studies using data read directly from CRSP stock databases or pre-extracted from any source. Eventus can also read raw returns, prices, bid and ask quotations, trading volume, number of trades and shares outstanding from the CRSP database. Eventus can print the raw data or store them in several disk file formats. The Eventus system includes utility programs to convert calendar dates to CRSP trading day numbers, convert CUSIP identifiers to CRSP permanent identification numbers, and extract event study cumulative or compounded abnormal returns for cross-sectional analysis.

Eventus provides user control over estimation periods and cumulative return windows, a choice of raw, comparison period mean adjusted, market adjusted or market model abnormal returns. Simple statements allow the researcher to run a complete event study, from reading the CRSP stock database to printing results, with a program as short as four lines.

This manual is not intended to be a textbook on event study methods, although citations to relevant literature are provided whenever possible. For an overview of event study methods, please see Peterson (1989), Binder (1998), and MacKinlay (1997). For a rigorous justification of standard event-study procedures, see Prabhala (1997).

Please visit our Web site, www.eventstudy.com, for additional usage examples, frequently asked question lists (FAQs), upgrade announcements, documentation and software sales and technical support.

Eventus is analytical software and does not include data. While Eventus can deal with all the CRSP stock databases mentioned in this manual, your institution or firm may not subscribe to all of them. If not, some features of

Eventus may not be of use to you. Users can supply stock and index returns from non-CRSP sources as well.

The symbols that appear in the statement descriptions in this manual have the following meanings. A vertical bar (|) indicates that only one of the words it joins may be chosen. Anything within square brackets ([]) is optional. **Boldface type** indicates that you enter the word exactly as it appears in the statement description. Replace a word or symbol set in *slanted Roman type* with a word or numeral of your choice. An ellipse (. . .) indicates that you can continue with additional specifications. Within the text, Eventus statement names and variable names are set in `typewriter style`. `EVENTUS` refers to the statement of that name, while Eventus refers to the software.

Chapter 2

Event Studies: The Essentials

> *"Much has been learned from the body of research based on event study methodology.... As one moves forward, it is expected that event studies will continue to be a valuable and widely used tool in economics and finance."* MacKinlay (1997, p. 38)

Eventus offers many options that let you tailor event study output to your needs. To start with, though, you might run an Eventus program in the simplest possible form, with almost no options. This may help you better understand the basic structure of the program.

Figure 2.1 shows the minimum set of statements needed to run an event study. The user can type the statements into the SAS Editor window for interactive submission (all at once or one line at a time), or create and save a separate program file (in plain EBCDIC text format on IBM mainframes and plain ASCII text on all other systems) for batch submission. Except for items inside quotation marks, the program statements are not case-sensitive. The example assumes the use of a CRSP daily stock database. Later chapters describe the usage of monthly CRSP data and non-CRSP data.

The `filename request` statement points to the *request file* prepared by the researcher.[1] The request file is a separate file that the researcher creates to define the sample for the study. Each line of the request file should contain a five digit PERMNO identifier and a date in the form CCYYMMDD. Later

[1] `filename` is a base SAS statement; this one defines the *fileref* `request`. A SAS fileref is a simple reference label associated with an external file. An external file means a file not in a format exclusive to SAS. Depending on the operating system, filerefs may be strictly internal to SAS , as when they are defined by `filename` statements, or they can be defined by operating system DDnames, logicals or environment variables.

Figure 2.1
The simplest Eventus event study program.

```
filename request 'F:\Any Folder\Filename.extension';
eventus;
 request;
 evtstudy;
```

Figure 2.2
Request file: PERMNOs and dates, illustrating tolerated irregularities.

```
72100     19970626
75111   19980209
77142     981016
36150     19970616
77170     19981118
75241     19980805
76263     19980210
76369 19970814
77446     19981117
83447     19980615
10506     19971013
   67652     19980330
91732     19980708
79739    970617
76754     19980615
77833     19970331
10914     19970306
```

chapters describe how to use CUSIP and similar identifiers instead of PERMNO. The date will be used as "day 0". Spacing is unimportant as long as at least one blank separates the PERMNO and date. For example, Figure 2.2 displays a request file that is perfectly acceptable, even though it includes irregular spacing and a mixture of two digit and four digit years.[2] The request file need not be sorted in any particular order.

The `EVENTUS` statement gets the package started. Here, the statement consists simply of the word `EVENTUS` and a semicolon. Later chapters describe options that can be used on the statement. The `REQUEST` statement in the

[2] Eventus interprets two digit years from 00–24 as 2000–2024, and two digit years from 25-99 as 1925–1999.

next line tells Eventus to read the *request file*. In the last line, `EVTSTUDY` produces the actual event study.

After the user submits all the statements, the results appear in the SAS Output window (or the procedure output file in the case of a batch run.) Figures 2.3 and following present the results for the above request file and Eventus statements. The first page of output summarizes the sample and some default settings. The second page presents a firm-by-firm listing of the sample, indicating how many daily returns Eventus found for each event. The firm name comes from the name history array of the CRSP database and is the name in effect on the event date. By default, Eventus produces portrait oriented output and abbreviates the name for page formatting reasons. If a share class letter exists in the database, it is reported as the last character of the abbreviated name column in the sample listing. In Figure 2.4, the name of Integrated Brands is truncated to make room for share class letter A. Full firm names can be printed by using landscape orientation, described in the next chapter.

The third page presents firm-by-firm and sample mean and median statistics of returns and estimated market model parameters. By default, the market model is estimated by ordinary least squares, using all available data from a 255 trading-day estimation period ending 46 trading days before the event date. In Figure 2.5, the alpha, beta and residual standard deviation are the estimated market model intercept, slope and root mean squared error, respectively.

The final three pages of output present the event study results. The event period is defined by default as from 30 days before through 30 days after the event date, and is broken into three "windows" for abnormal return cumulation: the pre-event period, days -30 through -2; days -1 and 0, which is the period most commonly examined for the immediate impact of the event; and the post-event period, days $+1$ through $+30$. The researcher can select different event period and window definitions using options described in later chapters. By default, Eventus reports three alternative return benchmark and standard error combinations. Figure 2.6 presents market adjusted returns, which are simple differences between the stock return and the market index return that do not use the market model. Figure 2.7 presents one set of market model results. The t-statistics in Figures 2.6 and 2.7 use standard errors estimated from the time series of sample mean (portfolio) abnormal returns; this approach avoids biases due to cross-correlated abnormal returns at the expense of somewhat reduced power. Figure 2.8 displays the

Figure 2.3
Sample Eventus output: First page.

```
          Eventus (R) Software from Cowan Research, L.C.                1


Eventus (R) software is produced by Cowan Research, L.C.
http://www.eventstudy.com/


ESTIMATION PERIOD:  Ends 46 days before the event date;
                        255 days in length.

TOTAL NUMBER OF EVENTS:         17
  EVENTS WITH USEABLE RETURNS:  16
  EVENTS DROPPED:               1


STATISTICAL SIGNIFICANCE LEVELS:  1 tailed

NONPARAMETRIC TEST:  Generalized sign test in all event studies.
For nonparametric tests, significance levels of .10, .05, .01 and .001
are denoted by (, <, <<, <<< or ), >, >>, >>> respectively.  Left
brackets -- (, < -- appear when the ratio of positive to negative
is less than in the parameter estimation period.  Right brackets
mean that the ratio is more positive than in the estimation period.


NOTE:  Useable returns means all nonmissing returns except the
      first day after a missing estimation period return.
```

market model results with a Z-test using the standardized abnormal return approach, which estimates a separate standard error for each firm-event and assumes cross-sectional independence. The default output also includes a nonparametric generalized sign test, which uses the fraction of positive abnormal returns in the event period as the "normal" instead of assuming 0.5. Appendix A presents further details of the methods.

Figure 2.4
Sample Eventus output: Second page.

Eventus (R) Software from Cowan Research, L.C. 2

Results of Daily Stock Returns Input

PERMNO	Name on Event Date	Event Date	Estimation Period Returns <=255	Event Period Returns <=61	Reason if no useable returns
10506	INTEGRATED BRAN A	10/13/1997	255	61	
10914	ALLWASTE INC CO	03/06/1997	255	61	
36150	CORE INDUSTRIES I	06/16/1997	255	61	
67652	AMERIWOOD INDUSTR	03/30/1998	255	54	
72100	IMO INDUSTRIES IN	06/26/1997	255	61	
75111	AMAX GOLD INC RP	02/09/1998	255	61	
75241	PIONEER NATURAL R	08/05/1998	253	61	
76263	ILLINOIS CENTRAL	02/10/1998	255	61	
76369	HEALTH MANAGEMENT	08/14/1997	255	61	
76754	BAY NETWORKS INC	06/15/1998	255	61	
77142	GREYHOUND LINES I	10/16/1998	255	61	
77170	CHRYSALIS INTERNA	11/18/1998	255	60	
77446	ESKIMO PIE CORP	11/17/1998	255	61	
77833	PEAK TECHNOLOGIES	03/31/1997	255	61	
79739	SEDA SPECIALTY PA	06/17/1997	255	59	
83447	EXCEL COMMUNICATI	06/15/1998	255	61	
91732	P M C INTERNATION	07/08/1998	*	*	No data on date

* No useable returns found. ** Or beyond estimation period.

Figure 2.5
Sample Eventus output: Third page.

Eventus (R) Software from Cowan Research, L.C. 3

Parameter Estimates and Estimation Period Statistics

---------- Index Weight=Equal ----------

PERMNO	Alpha	Beta	Mean Return	Market Model Residuals>0	Residual Standard Deviation
10506	0.00033	1.13	0.00192	42.74%	0.07700
10914	-0.00056	0.65	0.00021	41.56%	0.02483
36150	-0.00019	0.75	0.00043	42.74%	0.01934
67652	-0.00358	1.44	-0.00206	50.19%	0.02716
72100	-0.00265	1.12	-0.00192	51.76%	0.03949
75111	-0.00469	0.76	-0.00367	56.86%	0.03000
75241	-0.00310	1.27	-0.00137	45.45%	0.01967
76263	-0.00093	1.14	0.00056	45.49%	0.01278
76369	-0.00627	1.68	-0.00510	51.76%	0.10974
76754	-0.00287	2.87	0.00203	46.27%	0.03338
77142	0.00064	1.04	0.00109	43.13%	0.02837
77170	-0.00141	1.04	-0.00168	48.62%	0.07034
77446	-0.00041	0.62	-0.00058	45.49%	0.04138
77833	-0.00467	2.05	-0.00186	53.33%	0.05053
79739	-0.00093	1.20	-0.00003	47.45%	0.02780
83447	-0.00140	2.25	0.00244	42.35%	0.03508
MEAN	-0.00204	1.31	-0.00060	47.20%	0.04043
MEDIAN	-0.00141	1.13	-0.00030	45.88%	0.03169

Figure 2.6
Sample Eventus output: Fourth page.

Market Adjusted Returns, EW Index

Day	Average Abnormal Return	Median Abnormal Return	t	N	Positive: Negative	Generalized Sign Z
-14	0.69%	-0.17%	0.58	16	7:9	0.04
-13	-1.20%	-1.79%	-1.01	16	5:11	-0.97
-12	-0.48%	-0.70%	-0.40	16	7:9	0.04
-11	0.29%	-0.36%	0.24	16	6:10	-0.46
-10	-0.39%	-0.31%	-0.32	16	7:9	0.04
-9	-1.74%	0.13%	-1.46$	16	8:8	0.55
-8	6.47%	0.34%	5.42***	16	8:8	0.55
-7	-0.61%	-0.07%	-0.51	16	8:8	0.55
-6	0.88%	-0.18%	0.74	16	7:9	0.04
-5	-3.07%	-0.85%	-2.57**	16	5:11	-0.97
-4	0.99%	0.12%	0.83	16	9:7	1.05
-3	-2.47%	-0.36%	-2.07*	16	7:9	0.04
-2	4.22%	0.76%	3.54***	16	8:8	0.55
-1	4.74%	3.96%	3.97***	16	11:5	2.06>
0	18.25%	13.34%	15.30***	16	12:4	2.57>>
+1	0.07%	0.20%	0.06	16	8:8	0.55
+2	2.10%	-0.43%	1.76*	16	7:9	0.04
+3	-2.29%	-0.55%	-1.92*	16	4:12	-1.47(
+4	-2.39%	-0.49%	-2.00*	16	5:11	-0.97
+5	-1.56%	0.22%	-1.31$	16	9:7	1.05
+6	2.29%	-0.03%	1.92*	16	8:8	0.55
+7	-0.70%	0.00%	-0.59	16	8:8	0.55
+8	-0.27%	-0.30%	-0.22	16	7:9	0.04
+9	-1.84%	-0.56%	-1.54$	16	4:12	-1.47(
+10	-0.84%	-0.10%	-0.70	16	8:8	0.55
+11	5.33%	-0.08%	4.47***	16	6:10	-0.46
+12	-1.37%	-0.35%	-1.14	16	5:11	-0.97
+13	-1.55%	-0.03%	-1.30$	16	7:9	0.04
+14	-1.14%	-0.01%	-0.95	16	8:8	0.55

Days	Cumulative Average Abnormal Return	Median Cumulative Abnormal Return	t	Positive: Negative	Gen Sign Z
(-30,-2)	11.56%	11.85%	1.80*	11:5	2.06>
(-1,0)	22.98%	14.47%	13.63***	11:5	2.06>
(+1,+30)	5.82%	0.89%	0.89	10:6	1.56)

$, (,) significant at .10 *, <, > significant at .05
, <<, >> significant at .01 *, <<<, >>> significant at .001

Figure 2.7
Sample Eventus output: Fifth page.

Eventus (R) Software from Cowan Research, L.C. 5

Market Model, EW Index

Day	Average Abnormal Return	Median Abnormal Return	t	N	Positive: Negative	Generalized Sign Z
-14	1.13%	0.30%	0.95	16	10:6	1.23
-13	-0.73%	-1.39%	-0.61	16	6:10	-0.78
-12	-0.38%	-0.60%	-0.32	16	7:9	-0.28
-11	0.52%	-0.08%	0.44	16	8:8	0.22
-10	-0.02%	-0.16%	-0.02	16	7:9	-0.28
-9	-1.39%	0.40%	-1.17	16	9:7	0.72
-8	6.76%	0.92%	5.67***	16	11:5	1.73>
-7	-0.38%	0.00%	-0.32	16	8:8	0.22
-6	0.96%	0.16%	0.81	16	9:7	0.72
-5	-2.90%	-0.70%	-2.43**	16	5:11	-1.28
-4	1.22%	0.14%	1.03	16	11:5	1.73>
-3	-2.08%	-0.04%	-1.74*	16	8:8	0.22
-2	4.65%	1.04%	3.90***	16	9:7	0.72
-1	5.16%	4.08%	4.33***	16	12:4	2.23>
0	18.86%	15.08%	15.83***	16	12:4	2.23>
+1	0.19%	0.26%	0.16	16	8:8	0.22
+2	2.13%	-0.29%	1.79*	16	6:10	-0.78
+3	-2.01%	-0.33%	-1.69*	16	4:12	-1.78<
+4	-2.19%	-0.26%	-1.84*	16	7:9	-0.28
+5	-1.46%	0.40%	-1.23	16	10:6	1.23
+6	2.46%	-0.18%	2.07*	16	7:9	-0.28
+7	-0.63%	0.33%	-0.53	16	9:7	0.72
+8	-0.10%	-0.06%	-0.09	16	8:8	0.22
+9	-1.63%	-0.66%	-1.37$	16	5:11	-1.28
+10	-0.67%	0.19%	-0.56	16	8:8	0.22
+11	5.36%	-0.02%	4.49***	16	8:8	0.22
+12	-1.27%	-0.18%	-1.06	16	6:10	-0.78
+13	-1.33%	0.10%	-1.12	16	10:6	1.23
+14	-1.01%	0.12%	-0.85	16	11:5	1.73>

Days	Cumulative Average Abnormal Return	Median Cumulative Abnormal Return	t	Positive: Negative	Gen Sign Z
(-30,-2)	18.78%	11.99%	2.93**	11:5	1.73>
(-1,0)	24.02%	17.12%	14.25***	11:5	1.73>
(+1,+30)	11.41%	4.20%	1.75*	12:4	2.23>

$, (,) significant at .10 *, <, > significant at .05
, <<, >> significant at .01 *, <<<, >>> significant at .001

Figure 2.8
Sample Eventus output: Final page.

Market Model, Standardized Residual Method, EW Index 6

Day	Average Abnormal Return	Median Abnormal Return	Z	N	Positive: Negative	Generalized Sign Z
-14	1.13%	0.30%	1.59$	16	10:6	1.23
-13	-0.73%	-1.39%	-0.84	16	6:10	-0.78
-12	-0.38%	-0.60%	-0.18	16	7:9	-0.28
-11	0.52%	-0.08%	1.23	16	8:8	0.22
-10	-0.02%	-0.16%	0.22	16	7:9	-0.28
-9	-1.39%	0.40%	0.29	16	9:7	0.72
-8	6.76%	0.92%	3.58***	16	11:5	1.73>
-7	-0.38%	0.00%	-0.25	16	8:8	0.22
-6	0.96%	0.16%	1.42$	16	9:7	0.72
-5	-2.90%	-0.70%	-0.80	16	5:11	-1.28
-4	1.22%	0.14%	1.32$	16	11:5	1.73>
-3	-2.08%	-0.04%	-0.25	16	8:8	0.22
-2	4.65%	1.04%	2.63**	16	9:7	0.72
-1	5.16%	4.08%	4.70***	16	12:4	2.23>
0	18.86%	15.08%	19.82***	16	12:4	2.23>
+1	0.19%	0.26%	2.43**	16	8:8	0.22
+2	2.13%	-0.29%	0.79	16	6:10	-0.78
+3	-2.01%	-0.33%	-1.15	16	4:12	-1.78<
+4	-2.19%	-0.26%	-1.44$	16	7:9	-0.28
+5	-1.46%	0.40%	-1.41$	16	10:6	1.23
+6	2.46%	-0.18%	1.50$	16	7:9	-0.28
+7	-0.63%	0.33%	-0.05	16	9:7	0.72
+8	-0.10%	-0.06%	0.94	16	8:8	0.22
+9	-1.63%	-0.66%	-1.02	16	5:11	-1.28
+10	-0.67%	0.19%	0.57	16	8:8	0.22
+11	5.36%	-0.02%	2.15*	16	8:8	0.22
+12	-1.27%	-0.18%	-0.56	16	6:10	-0.78
+13	-1.33%	0.10%	-0.19	16	10:6	1.23
+14	-1.01%	0.12%	-0.37	16	11:5	1.73>

Days	Cumulative Average Abnormal Return Equally Weighted	Precision Weighted	Median Cumulative Abnormal Return	Z	Positive: Negative	Generalized Sign Z
(-30,-2)	18.78%	16.18%	11.99%	3.92***	11:5	1.73>
(-1,0)	24.02%	18.81%	17.12%	17.34***	11:5	1.73>
(+1,+30)	11.41%	8.39%	4.20%	1.94*	12:4	2.23>

$, (,) significant at .10 *, <, > significant at .05
, <<, >> significant at .01 *, <<<, >>> significant at .001

Chapter 3

Event Studies: The Options

Eventus lets you configure many details of your event study to suit you. This chapter tells how.

3.1 Event Studies Centered on a Single Event Date

The typical event study analyzes returns around one type of event date at a time. If you need to cumulate returns between two paired event dates, when the number of periods between them varies from firm to firm, see section 3.3 after reading this section. Figure 3.1 displays the Eventus statements to run an event study with a single event date. The options are described below.

The EVENTUS statement

Use options on the EVENTUS statement when you need to provide Eventus with special information about the CRSP stock database to be used and to set the orientation of printed output. By default, Eventus assumes that you are using a CRSP daily database.

MONTHLY To use the monthly CRSP database instead of the default daily, specify MONTHLY on the EVENTUS statement. For example,

```
EVENTUS MONTHLY;.
```

Figure 3.1

Eventus statements for an event study centered around a single date.

filename request 'G:\Some Folder\Filename.extension';

EVENTUS [**MONTHLY**|**EXCESS**] [**ESTINTER=DAY**|**MONTH**]
[**PAGE=WIDE**];

[**TITLE** *'text'*;]

[**TITLE2** *'text'*;]

REQUEST [**CUSIP**|**CUSIPERM**] [**ID=***variable* **IDFMT=***format*]
[**GROUP=***variable*] [**GRWEIGHT**]
[**DATEFMT=MMDDYY**|**YYMMDD**|**DDMMYY**|**DATE**|**CRSP**]
[**AUTODATE**[**=BACK**]]
[**NODIVIDX**] [**SP500**|**COMPOSIT**|**SIZEINDX**[**=CRSP**]]
[**EST=**−*value*| +*value*] [**POOL**] [**ESTLEN=***n*]
[**MINESTN=***n*] [**SHORT**] ;

WINDOWS [*(begin,end) (begin,end) ...*];

EVTSTUDY [**NONAMES**|**NOPLIST**|**SIC**]
[**DETAIL**|**DETAIL=FULL**] [**RAW**] [**CP**] [**NOMAR**]
[**NOMM**|**NOSTD**|**STDONLY**|**STDALL**]
[**VALUE**|**BOTH**|**SPORT**] [**SW**|**GARCH**|**EGARCH**]
[**TIMEUNIT=***periods*] [**PRE=***periods*] [**POST=***periods*]
[**OVERLAP**] [**MAXMISS=***n*]
[**CSECTERR**] [**STDCSECT**|**EGLS**|**CDCSI**] [**SERIAL**]
[**RANKTEST**|**JACKNIFE**] [**BOOT**] [**MEDIAN**] [**BUYHOLD**]
[**TAIL=1**|**2**] [**BTAIL=1**|**2**] [**OUTSAS=***libref.membername*]
[**PACKAGE=***specifier*];

EXCESS This option requires a separate add-on subscription to the CRSP Excess Returns or CRSP Index File/Portfolio Assignments File. With CRSPAccess format and the add-on subscription installed, the database includes risk class decile numbers for individual stocks that Eventus uses to select appropriate size-decile portfolio returns. To do this, the plain text (ASCII or EBCDIC character) versions of the Indices subscription files must be accessible. Specifically, Eventus needs the files `dsbo.dat` and `dsbc.dat` to be associated with the SAS filerefs `statidx1` and `statidx2`. This is done by submitting from the Editor window, or including in a batch program file, statements similar to the following, where `/crsp/indices/` should be changed to the full Unix, Windows or OpenVMS specification for the directory or folder containing the character indices files.

```
filename statidx1 '/crsp/indices/dsbc.dat';
filename statidx2 '/crsp/indices/dsbo.dat';
```

The `EXCESS` option indicates that the beta-matched or standard-deviation matched (see the `SPORT` option) decile portfolio return is to be used as the market index. The results labeled "market adjusted returns" correspond to what were traditionally called "CRSP Excess Returns." However, the risk-matched decile portfolio returns are used with all event study methods. For example, with the `EXCESS` option, "market model" event study results use risk-matched decile portfolio returns in place of a broad market index, and the returns statement with the index option reports raw stock returns with the risk-matched decile portfolio return labeled as the index return.

ESTINTER The `ESTINTER` option indicates that the event study will use two different return intervals, one for the estimation of benchmark model parameters and one for the event period. For example, to conduct a daily-return event study using market model parameters estimated from monthly returns, specify `ESTINTER=MONTH`. No option is needed to specify the return interval for the event period in this illustration, because daily is the default return interval. To use the `ESTINTER` option with CRSP data, you must subscribe to a CRSP stock database with the desired estimation period and event period return-computation intervals. To run a daily event study with monthly parameter estimation, you need both daily and monthly CRSP stock databases. When working with a CRSP SFA (legacy) format database, a `filename crspes` statement should point to the main stock data file and a

`filename estindx` statement should point to the calendar/index file. On a system that permanently stores the CRSP data in SAS data sets, the option `ELIBNAME=libname` should appear on the `EVENTUS` statement to point to the aggregate storage location (typically a folder or directory) containing the CRSP data with the frequency to be used for the estimation period. When working with CRSPAccess format databases, Eventus uses the CRSPAccess environment variables or logicals to locate the data.

`PAGE=WIDE` Normally Eventus prints portrait-oriented, or tall, pages. The default output is easy to view on a display screen before printing, and is suitable for almost any printer. The `PAGE=WIDE` option on the `EVENTUS` statement allows Eventus to print landscape-oriented, or wide, output. With wide output, more information will fit on one page. For example, Eventus will print the complete firm names from the CRSP database for the stocks in the sample, the day of the week of the event dates, and up to 95 days of abnormal returns in the event period.[1] Wide output produces a line width of 132 characters. It may be necessary to manually select landscape orientation when printing it.

The `REQUEST` statement

The `REQUEST` statement instructs Eventus to read the request file and the calendar of the CRSP database, then match the dates in the request file to CRSP dates. The request file is an external file; you use it to supply the PERMNO identifiers for your sample, the event dates, and depending on the options you specify (see below), possibly other information. Chapter 2 describes the basic structure of the request file.

For monthly event studies, the dates in the request file can be any day of the month. For weekly event studies, use the last trading day in the week, or specify any day up to the last trading day and use the `AUTODATE` option described below.

The rest of this section describes `REQUEST` statement options for the processing of dates and market and stock returns and for the construction of the estimation period. A SAS `filename` statement or the operating system must

[1] The `ALLDAYS` option (see page 28 below) is available with either wide or tall output to force Eventus to use additional pages to avoid the limitation on the number of days.

associate the fileref `REQUEST` with the request file.[2]

Option to convert CUSIPs to PERMNOs

If you have CUSIPs in the request file, you can specify the `CUSIPERM` option to have Eventus internally match them to the corresponding CRSP PERMNOs but not alter the request file. To create a permanent request file with CUSIPs replaced by PERMNOs, please see Chapter 9.

The `CUSIPERM` option works properly only if the PermnoUp program described in the Eventus installation instructions is run during Eventus installation and after each annual or quarterly update of the CRSP database. Only those CUSIPs present in the CRSP database will be matched.

If you save the event study results in a permanent SAS data set (see the `EVTSTUDY` statement options below), only the PERMNOs will be included in the file.

Including an identification variable

Each line of your request file may include an identification variable following the event date. Eventus will read it, and print it on subsequent output, if you name it with `ID=`. `ID=ID` works, or you can choose another name. Use `IDFMT` to tell Eventus what format to use for reading and printing the identifying variable. `IDFMT=4` means a 1–4 digit integer, while `IDFMT=$4` means a four letter word. Other lengths and other SAS formats are permitted also.

Grouping variables and group weights

When you specify `GROUP=`*variable name*, Eventus looks for a grouping variable in the request file. Replace *variable name* with a valid SAS name, for example `GROUPID`. The grouping variable should follow the event date, and the `ID` variable if there is one, on each line of the request file. The grouping variable must be an integer between 0 and 9999 inclusive. Leading zeros are optional.

If any two or more observations have the same grouping variable value, Eventus combines the observations into an event time portfolio and treats the portfolio as a single observation in the event study. An observation with a unique value of the grouping variable is treated normally.

[2] To use a SAS data set as the request file, please see the `INSAS` option on page 147.

By default, the observations with a common grouping variable value receive equal weights in the group portfolio. To change this, specify the `GRWEIGHT` option. Each line of the request file then must contain a within-group weight on each line after the grouping variable value. The weights must sum to 1.

For example, suppose a researcher specifies this `REQUEST` statement:

```
request id=Event group=GROUP grweight;
```

The following request file provides the required information.

```
56899 19850131 00 84 .08333
56899 19860131 01 85 .16666
56899 19870211 02 86 .20
56899 19880205 03 87 .20
60038 19850227 04 84 .08333
60038 19860226 05 85 .16666
60038 19870302 06 86 .20
60038 19880223 07 87 .20
28804 19850130 08 84 .08333
28804 19860314 09 85 .16666
28804 19870206 10 86 .20
28804 19880209 11 87 .20
42083 19841018 12 84 .25
42083 19851016 13 85 .16666
42083 19861016 14 86 .20
42083 19871022 15 87 .20
18817 19850213 16 84 .25
18817 19860307 17 85 .16666
18817 19870310 18 86 .20
18817 19871221 19 87 .20
44740 19850222 20 84 .25
44740 19860225 21 85 .1667
```

Following the PERMNO and event date, each line lists the identifying number Event, the group number GROUP, and the weight for the observation within its group. (The example happens to have several repeated PERMNOs, but this has no particular relation to the grouping feature — there could just as well have been 21 unique PERMNOs.) Event 00 is part of GROUP 84 and is to receive weight .08333, while Events 04, 08, 12, 16 and 20 in the

same group receive weights of .08333, .08333, .25, .25, and .25 respectively. Other observations are part of other groups. Notice that the groups need not all have the same number of members. Had the `GRWEIGHT` option not appeared on the `REQUEST` statement, the weight information would have been ignored. Each observation in a group portfolio would have received equal weight. Thus, the researcher could check the sensitivity of the results to the group weighting by changing only one option.

Creating arbitrage portfolios by specifying stocks to sell short

Each stock normally has a positive weight in the event-time portfolio. To indicate that some stocks should enter the portfolio as if they are sold short, specify `SHORT` on the `REQUEST` statement. In the request file, place either an `S` or an `L` as the last item of each line to indicate short or long. When an `S` appears, Eventus reverses the sign of each estimation period and event period stock return for that observation. The sign reversal occurs just after the return is read from the CRSP database and verified as non-missing. Thereafter, Eventus treats the returns the same as the returns for any other observation. The same stock can be held short in one line of the request file and long in another; Eventus re-reads the stock returns from the CRSP database each time, even if the dates are the same.

Note that Eventus does not make any explicit adjustment to the portfolio weights — the calculations still treat the sample as an equally weighted portfolio (precision weighted portfolio in the case of standardized abnormal return tests) of stocks held long. (That is, after performing the sign reversal, Eventus makes no further distinction between `S` and `L` stocks.) This allows you to create an arbitrage portfolio by specifying a short position for half the observations and a long position for the other half. The portfolio weights sum to zero in an arbitrage portfolio.

Studies of insider trading often use combinations of short and long positions in abnormal return tests. For examples, see Rozeff and Zaman (1988) and Arshadi and Eyssell (1991).

Options for processing dates

The `REQUEST` statement allows you to specify how Eventus should handle the dates in your request file.

`DATEFMT=`

Calendar dates You may list calendar dates in nearly any conventional format. The default is YYMMDD, which automatically accommodates both eight digit (four digit year) and six digit (two digit year) dates. Besides MMDDYY and DDMMYY, you can use the SAS date format DATE.[3]

CRSP Trading Day Numbers Eventus never requires you to manually convert calendar dates to CRSP day numbers. If you have CRSP day (or week or month) numbers, you can use them in your request file. Specify `DATEFMT=CRSP` on the `REQUEST` statement.

`AUTODATE` If some of the calendar dates in the request file are potential non-trading days, you have two choices. Your first choice is to run your event study normally. Eventus will tell you which observations have non-trading dates. You can then make the necessary corrections and re-run the event study to have the observations included. Your second choice is to specify `AUTODATE` on the `REQUEST` statement. `AUTODATE` tells Eventus to convert automatically *all* calendar dates to trading days. Non-trading days are converted to the following trading day. For example, a Saturday would be changed to the following Monday, or Tuesday if Monday were a holiday. To convert non-trading dates to the previous trading date instead of the next, specify `AUTODATE=BACK`.

Market index options

`NODIVIDX` Eventus normally uses the returns including dividends of the basic equally weighted and value weighted indexes in the CRSP database. Specify `NODIVIDX` to instruct Eventus to use the index returns excluding dividends.

`SP500` **and** `COMPOSIT` The CRSP NYSE-AMEX-Nasdaq database reports the Standard and Poor's 500 Composite Index in addition to the equally weighted and value weighted indexes of all stocks. Specify `SP500` to tell Eventus to read the Standard and Poor's index return instead of the value weighted index. A CRSP SFA database containing only Nasdaq data may report the Nasdaq Composite Index return instead of the Standard and Poor's index. To use

[3] An example of a date in DATE format is 19OCT1987.

the Nasdaq Composite Index instead of the value weighted index, specify `COMPOSIT`. (Eventus does not determine whether the CRSP database contains NYSE-AMEX-Nasdaq, Nasdaq-only, or some other security universe; the `SP500` and `COMPOSIT` options are the same except for the way the output is labeled. It is up to you to make sure the database you use is the one you intended.)

`SIZEINDX` This option allows the use of size-portfolio returns. Eventus uses the size (capitalization) portfolio membership information on the CRSP stock file to determines which size portfolio return to use. All tests described in this manual are conducted as they would normally be, except that the index return is a size portfolio return rather than the return on a broader market index. A SAS `filename` statement, operating system command, control language statement or environment variable must associate the fileref `sizeindx` with the size-portfolio index file. The size-portfolio index file must be in one of two specific formats, either the CRSP SFA ASCII character Indices/Decile File format or the Eventus size index file format. The Eventus size index file format is binary or character depending on whether the current program is using binary (SFA or CRSPAccess) or character files.[4] If the `sizeindx` fileref points to a CRSP Indices/Decile File, specify `SIZEINDX=CRSP` on the `REQUEST` statement. When a CRSP Indices/Decile File is used in conjunction with a CRSPAccess database, the specific Indices/Decile File to use depends on whether the CRSPAccess indices and portfolio assignments module was installed to integrate it with the main stock data. With the module installed, use `dsic.dat` or `msic.dat` for daily or monthly data, respectively, and specify `PORTYPE=2` on the `REQUEST` statement. Without the module installed, use `dsix.dat` or `msix.dat` for daily or monthly data, respectively, and do not use the `PORTYPE` option.

[4] In the Eventus binary format, the date expressed as a YYMMDD or CCYYMMDD integer occupies the first four bytes of each record. The ten size portfolio returns for that date follow in the next 40 bytes, each represented as a four-byte real value. For an Eventus character format file, each line begins with a single blank space, followed by the date as a six-digit YYMMDD or eight-digit CCYYMMDD integer, followed by a single blank space, followed by the first size portfolio return in scientific notation with six decimal places (E13.6 format); the remaining size portfolio returns follow, each separated by a single blank space. Note that portfolio 1 contains the stocks with the smallest capitalizations, while portfolio 10 contains the stocks with the largest capitalizations. A program to generate an index file to these specifications is included with Eventus.

Computing continuously compounded returns

If you want to analyze returns in continuously compounded form, specify `LOG` on the `REQUEST` statement. Eventus then transforms each firm and index return r_{jt} to $\log_e(1 + r_{jt})$ as it is read from the CRSP database. By default, Eventus does not make the log transformation.

Options for constructing the estimation period

The estimation period is the time period used for running market model regressions, computing comparison period mean returns, and so on.

`EST=` and `POOL` By default, Eventus determines the estimation period for each firm by subtracting 46 trading days from the event date in your request file. The resulting date becomes the *last* day of the estimation period. If you want the estimation period moved back, say to 90 days before the event date, specify `EST=−90`. If you need an estimation period following the event date, specify a positive number. `EST=+61` (the plus sign is optional) gets you an estimation period that *begins* on day +61.[5]

The default estimation period for weekly event studies ends with week −10; for monthly event studies, the default is to end the estimation period with month −13.

If you need to split your estimation period between pre- and post-event dates, specify `POOL` on the `REQUEST` statement. Then Eventus will chop your estimation period into two equal halves. For example,

```
REQUEST ...  EST=50 POOL;
```

gets you an estimation period of which the first half ends with day −50, and the second half begins on day +50.

`EST` has no effect on the length of the estimation period. The `ESTLEN` option, described next, changes the estimation period length.

`ESTLEN=` By default, the estimation period is 255 days long when using daily data and 52 weeks or 60 months long when using weekly or monthly data. You can change the estimation period length with `ESTLEN`. The largest

[5] It is also possible to specify the estimation period by calendar date or trading day or month number; see `EST=SPECIFIC` on page 146 in Appendix B.

number you can use is 999 and the smallest is 3. Eventus assumes that you are specifying the number of months for a monthly event study, weeks for a weekly event study, or days for a daily event study.

`MINESTN=` Specify `MINESTN=`*n* to remove an observation from the sample if the stock has fewer than *n* days or months of return data in the estimation period. For example, `MINESTN=60` means that if the stock has fewer than 60 usable returns in the estimation period, that observation will be dropped. Eventus tells you when observations drop out of the sample. By default, the only constraint on estimation period data is the requirement of three returns for running a market model regression. Don't specify a `MINESTN` greater than `ESTLEN` or you won't have any events left at all!

The `WINDOWS` statement

The `WINDOWS` statement is optional. Use it to specify ranges of dates — "windows" — in event time over which Eventus should report cumulative or compounded abnormal returns and associated test statistics. Each window specification requires a *begin* and *end*, which can be the same if desired. For example, (−3,0) specifies a four day or four month window ending with the event date; (2,2) specifies a one day or one month window containing the second day or month after the event date. When *begin* and *end* are equal, no cumulation or compounding occurs; the only purpose for such a specification would be to cause a single date to appear in the window output section for ease of reference. If the `WINDOWS` statement is omitted, Eventus reports three windows: (−`PRE`,−2), (−1,0) and (+1,+`POST`). The results for the windows appear at the bottom of the same page on which the results for individual dates are printed, or on a separate page if there is not enough space.

The `EVTSTUDY` statement

The `EVTSTUDY` statement tells Eventus to run an event study. The options permit you to change the default event period, control the handling of observations with missing returns, select market indices and methods for computing abnormal returns and test statistics, increase or decrease the amount of printed output, and store results in a SAS data set.

Selecting the amount of printed output

EVTSTUDY normally produces a report of the results of searching the CRSP database for your sample. The report lists the PERMNO, identifying variable and event date from your request file, together with the name of the firm from the CRSP database. The report also tells, for each firm, how many returns Eventus found in the estimation period and in the event period. If there was a problem, for example if the date you asked for was outside the range of data for that firm, the report will state exactly what the problem was. If you prefer not to have Eventus print this report, specify NONAMES on the EVTSTUDY statement.

Unless you specify NOPLIST on the EVTSTUDY statement, you also get a list of the estimation period return statistics for each observation and the mean and median for the sample. The report includes market model alphas, betas and residual standard deviations, fraction of market model residuals positive, and additional statistics if space permits.

The last set of output from EVTSTUDY contains the event time results.

The SIC option causes Eventus to report SIC (industry classification) codes from the CRSP database on the list of estimation period return statistics. If an output SAS data set is being created (see the OUTSAS option below), the character variable SICCODE is included in the data set. The SIC code for the last reporting date on or before the event date is listed, unless the earliest reporting date precedes the event date; then the first SIC code is listed. See the CRSP release notes for information about the accuracy of the SIC code data.

By default, one page is generated for each abnormal return method in effect. Each page lists the portfolio mean abnormal return, test statistic, number positive and negative, and significance levels, for the event date ("day 0") and as many as 34 surrounding days (94 if PAGE=WIDE appears on the EVENTUS statement), depending on PRE and POST.[6] At the bottom of the page, for each window you request, are the portfolio cumulative or compounded abnormal return, the test statistics, the number positive and negative and significance symbols.

You may want to see window or daily abnormal returns for individual firms. EVTSTUDY will print individual firm abnormal returns only for the standardized abnormal return method. (The standardized abnormal return

[6] To print all the dates in the range defined by PRE and POST when the number exceeds 35 (or 95 with PAGE=WIDE), see the ALLDAYS option (page 28.)

method can be extended to non-market model benchmarks by specify the STDALL option (page 25.) To obtain window cumulative abnormal returns for every firm, specify DETAIL on the EVTSTUDY statement. As many firms as can fit will appear on a single page. To obtain complete daily abnormal returns for each firm, specify DETAIL= FULL. Full detail takes at least one page for each firm, so be prepared for lots of output. If you need to look at only a few individual days, use the following trick. Suppose you want to see day +1 for every firm. On the WINDOWS statement, include (1,1). Then specify DETAIL (*not* DETAIL= FULL) on the EVTSTUDY statement. This way you get the day you need without all the extra output. DETAIL specifications are ignored if you specify NOMM or NOSTD unless you also specify STDALL.

To save or export individual firm raw or abnormal returns for further analysis, use the OUTSAS option (page 31) to store intermediate results and also see the EXTRACT statement in Chapter 5.

Specifying benchmarks and standardized or non-standardized tests

The default is to compute three sets of abnormal returns and test statistics: market adjusted returns, market model abnormal returns using portfolio time-series standard errors and market model abnormal returns using standardized abnormal return tests. To add comparison period mean adjusted returns, specify CP on the EVTSTUDY statement. To add total unadjusted, or "raw", returns, specify RAW. To omit market adjusted returns, specify NOMAR. To omit market model returns altogether, specify NOMM. To omit just the standardized abnormal return tests using the market model, specify NOSTD. To omit everything *except* the market model with standardized abnormal return tests, specify STDONLY.

These options can be combined. For example, CP NOMAR NOMM causes EVTSTUDY to analyze comparison period mean adjusted returns, and omit all other types of abnormal returns.

To tell Eventus to report standardized abnormal return tests for non-market model methods, specify STDALL on the EVTSTUDY statement.

Specification of a market index

Normally, Eventus uses the CRSP equally weighted market index. Specify VALUE to change to the value weighted index or BOTH to produce a set of results using each.

Using standard deviation portfolio excess returns

The SPORT option applies only when EXCESS is specified on the EVENTUS statement. By default, when the EXCESS option is in effect and the appropriate add-on CRSP module is available, Eventus based on decile portfolios of stocks ranked by beta. Optionally, Eventus can use excess returns based on a ranking by standard deviation of return. To use standard deviation excess returns instead of beta excess returns, specify SPORT on the EVTSTUDY statement.

Scholes-Williams and GARCH market model estimation

Eventus reports market model results using both ordinary least squares and Scholes-Williams (1977) beta estimation when you specify SW.

When you specify the GARCH option, Eventus estimates the market model assuming a GARCH(1,1) error structure. For exponential GARCH, or EGARCH(1,1), specify the EGARCH option. With either the GARCH or EGARCH option, maximum likelihood estimates using the dual quasi-Newton algorithm are produced. Eventus reports the alpha and beta estimates as it does in the case of ordinary least squares, but does not report the estimated parameters of the conditional error variance model. No more than 40 iterations will be performed for each stock. In general, convergence will be better the longer the estimation period (ESTLEN) and better with the EGARCH(1,1) than the GARCH(1,1) model.

Cowan and Sergeant (1996) report that event study test specification and power are insensitive to the use of Scholes-Williams versus OLS estimation. Corhay and Rad (1996) and Brockett, Chen and Garven (1999) discuss the potential benefits of GARCH estimation in event studies. Bollerslev, Chou and Kroner (1992) provide an overview of GARCH, EGARCH and related models in finance.

Using multiperiod returns as the basic unit of analysis

To combine each consecutive *n* days, weeks or months into a single time unit for purposes of estimation and testing, specify TIMEUNIT=*n*. For example, in an event study using the CRSP daily stock database, TIMEUNIT=2 specifies that two-day returns are to be computed and used as if they were daily returns. [For an illustration of the technique, see the lower panel of Table VI in Bhagat, Marr and Thompson (1985).]

When n is even, period 0 in event time contains day 0 (the date in the request file), $\frac{n}{2} - 1$ days following day 0 and $\frac{n}{2}$ preceding day 0. Additional periods are formed on either side of period zero. For example, `TIMEUNIT=2` results in period zero containing days -1 and 0, period $+1$ contains days 1 and 2, and period -1 contains days -3 and -2. If n is odd, period 0 is centered on day zero. `TIMEUNIT=3` produces a period 0 containing days -1, 0 and $+1$; a period $+1$ containing days $+2$, $+3$ and $+4$, and so on.

Eventus combines daily, weekly, or monthly returns into period returns by addition. If the `LOG` option is specified on the `REQUEST` statement, individual returns are converted to logarithmic form before adding.

When `TIMEUNIT` is specified, Eventus interprets the `WINDOWS` statement arguments and the `PRE`, `POST` and `MAXMISS` options or their defaults in terms of multiday, multiweek or multimonth periods. All results are reported in terms of multiday, multiweek or multimonth periods as well. However, Eventus interprets the `EST`, `ESTLEN`, `MINESTN` and other `REQUEST` statement options in terms of the original days, weeks or months. For example, in a daily event study, suppose the user specifies `ESTLEN=100` and `TIMEUNIT=2`. The market model and other estimates are computed on 50 two day returns — a total of 100 single days in the estimation period.

The use of `TIMEUNIT` precludes the examination of the returns of single actual days, weeks or months; Eventus converts each sequence of n returns to a multiday, multiweek or multimonth return as soon as it is read from the CRSP file. Another method of evaluating individual dates is to use the `WINDOWS` statement without `TIMEUNIT`. `WINDOWS` lets the user aggregate selected date ranges in event time for reporting and testing.

Specifying the number of days or months in the event period

`EVTSTUDY` computes and reports abnormal returns for the event period, which is defined by default as days or weeks -30 through $+30$, or months -12 through $+12$. To change from these defaults, specify the number of days before and after on the `EVTSTUDY` statement, using `PRE` and `POST`. These options may be used singly or in combination. For example, `PRE=60` means that the abnormal returns are to start with day -60; unless `POST=` is also specified, the latter retains its default value.

If the event period length (`PRE`+`POST`+1) exceeds 35 days, weeks or months, only 35 normally are printed (95 if the `PAGE=WIDE` option is in effect.) Any windows specified on a `WINDOWS` statement are printed, however,

regardless of the range of dates covered. If you want to print results for all the dates individually, specify ALLDAYS on the EVTSTUDY statement. For further discussion, see page 130 in Appendix B.

If you specify PRE= or POST=, you may need to change the estimation period from the default. To change the estimation period, specify EST= on the REQUEST statement. Eventus will stop and print an error message if the event period defined by PRE and POST overlaps the default or specified estimation period. If for some reason you want to override overlap checking and allow the estimation period and the event period to have dates in common, specify OVERLAP on the EVTSTUDY statement.

Dropping observations with missing event period returns

Specifying MAXMISS=*n* on the EVTSTUDY statement causes Eventus to exclude from the analysis any firm that has more than *n* missing returns in the event period (−PRE,+POST). To keep only those observations with no missing event period returns, specify MAXMISS=0.

Computing cross-sectional standard errors instead of time series standard errors

Normally Eventus computes portfolio standard errors from the time series of estimation period portfolio abnormal returns. The CSECTERR option specifies that the standard error for each date in event time should be computed across securities instead. For an example of the application of the cross-sectional method, see Pilotte (1992). This option has no effect on the standardized abnormal return method.

The standardized cross-sectional test

The z statistic that Eventus normally computes for the standardized abnormal return method is the widely used statistic described by Patell (1976). The STDCSECT option substitutes the standardized cross-sectional test. This is an extension of the Patell test introduced by Boehmer, Musumeci and Poulsen (1991). The standardized cross-sectional test compensates for a possible variance increase on an event date by performing a simple cross-sectional variance adjustment. The SERIAL option explained below is automatically invoked by the STDCSECT option.

EGLS and Collins-Dent tests

The EGLS and CDCSI options replace the default standardized test with the estimated generalized least squares test and Collins-Dent test assuming cross-sectional independence, respectively. These tests are discussed in detail by Sanders and Robins (1991). The EGLS and CDCSI options automatically invoke the SERIAL option.

Adjusting window z tests for serial dependence

The SERIAL option applies to standardized abnormal returns only. Normally, the test statistics for abnormal returns cumulated over intervals you define in the WINDOWS statement are not adjusted for serial dependence. Mikkelson and Partch (1988) and others perform such a correction on cumulative returns. The SERIAL option causes Eventus to apply the correction to standardized abnormal returns only. Karafiath and Spencer (1991) and Cowan (1993) provide simulation evidence of the properties of the corrected and uncorrected test statistics.

Note that the serial dependence that the SERIAL option corrects is not due to any presumed dependence in the true market model error term, but occurs because all of the abnormal return estimators being cumulated are functions of the same estimators of the market model parameters. The derivation of the corrected standard error used by Mikkelson and Partch (1988) requires that the abnormal return be interpreted as a forecast error.

Alternative nonparametric tests

Normally, Eventus prints the generalized sign test statistic (see Cowan, 1992) for each day in the event period and for each window with both standardized and non-standardized tests. If you specify RANKTEST on the EVTSTUDY statement, Eventus instead will report the nonparametric rank test introduced by Corrado (1989) with non-standardized tests. The JACKNIFE option produces the jackknife test developed by Giaccotto and Sfiridis (1996). The RANKTEST and JACKNIFE tests are mutually exclusive. The generalized sign test still will be reported with standardized abnormal return method results.

Bootstrapped versions of parametric tests

The BOOT option on the EVTSTUDY statement produces bootstrapped versions of parametric tests. Eventus performs bootstrap tests only for the windows, not each individual day or month. However, you can obtain bootstrap tests for an individual day or month by specifying a window (on the WINDOWS statement) with the same beginning and ending date. The bootstrap tests do not replace other results in the output, but appear on a separate page after the regular parametric and nonparametric tests.

The BOOT option produces true bootstrap, or resampling, tests, using the approach described by Kramer (2000). Eventus does not compute out-of-sample bootstrap, more precisely called simulation, tests.

Eventus restricts the parametric test methods that it bootstraps to cross-sectional tests on buy-and-hold abnormal returns, and the standardized cross-sectional test on cumulative abnormal returns.

Reporting median abnormal returns

This option is needed only when the PAGE=WIDE option appears on the EVENTUS statement. With the default output format, PAGE=TALL, Eventus always reports median and mean abnormal returns. With the optional wide output format, Eventus reports median abnormal returns in place of the number positive and negative if you specify MEDIAN on the EVTSTUDY statement. Significance symbols for the default generalized sign test or optional rank test still are reported.

Computing buy-and-hold compounded window returns

To obtain buy-and-hold abnormal returns for multiperiod windows, specify BUYHOLD on the EVTSTUDY statement. The option applies only to non-standardized abnormal return methods. (By default, Eventus reports non-standardized results for each abnormal return method used in an event study, and standardized results in addition for the market model abnormal returns.) Eventus computes buy-and-hold abnormal returns by compounding successive daily (or other period) raw returns and market index returns, then adjusting the raw returns according to the abnormal return method used. Comparison period mean returns and market model alphas are adjusted for the window length. The Eventus output labels the mean buy-and-hold window returns

"Average Compounded Abnormal Return," while the default additive window abnormal returns appear as "Cumulative Average Abnormal Return."

Reporting one- instead of two-tailed tests

Eventus reports all significance levels for one-tailed tests by default. `TAIL=2` changes to two-tailed tests.

Saving results to a file

By default, none of the event study results is saved in a disk file. To store firm-by-firm results, specify the SAS file to store them in using the `OUTSAS=` option.

The *libref* parameter specifies the libref of the SAS data library in which to store the results. Eventus will create a new data set unless the one named already exists in the SAS data library, in which case the existing data set will be overwritten. A SAS `libname` statement or the operating system must associate the libref with the file or directory used as the library.

A related option, `PACKAGE=`, determines the content of the file. If you intend to use `OLDSTUDY` to reprint or merge event study results, specify `PACKAGE=1`. If you plan to use `EXTRACT` to create dependent variables for cross-sectional analysis of abnormal returns (see Chapter 5), specify `PACKAGE=DG`, or just `PACKAGE=D` if you do not need weights or standardized returns. Table B.1 on page 134 lists the complete set of `PACKAGE` specifications available.

3.2 An Event Study Example

O'Hara and Shaw (1990) conduct an event study of the statement by the U.S. Comptroller of the Currency in 1984 that some banks are too big to be allowed to fail. Figure 3.2 shows the Eventus program to replicate the study for the sample of 22 banks that were insolvent according to a measure proposed by Swary (1986). The `EVENTUS` statement needs no options for this study. O'Hara and Shaw state that their estimation period is days -55 through -6, relative to the event date of September 20, 1984. To replicate their estimation period we specify `EST=-6` and `ESTLEN=50` on the `REQUEST` statement. We identify the individual bank stocks using the numbers from

Figure 3.2
Eventus program for an event study of the "Too Big to Fail" policy.

```
filename request 'C:\Research\Banks\permnos.dat';
eventus;
 title ''O'Hara and Shaw, JF, Dec 1990, pp.1587-1600'';
 title2 ''Replication of Table III.C:  Insolvent Bank Sample'';
 request est=-6 estlen=50 id=Swarynum idfmt=2.0;
 evtstudy nomar nostd value post=5 pre=5 tail=2;
```

Swary's (1986) list, so we call the identifying variable `Swarynum`. The specification `IDFMT=2.0` means that the identifying values are integers of up to two digits.

O'Hara and Shaw (1990) report only market model abnormal returns; the option `NOMAR` on the `EVTSTUDY` statement suppresses the computation and reporting of market adjusted returns. Because the firms are all in the same industry and have a common event date, the authors do not use a standardized residual test, but base their test statistic "on a standard deviation estimated for the portfolio of sample firms from residual returns in the estimation period". [See Chandra and Balachandran (1990) and Chandra, Moriarity and Willinger (1990) for further discussion of cross-sectional dependencies in event studies.] We suppress standardized tests with the `NOSTD` option. The `NOMAR` and `NOSTD` options are not necessary; they only eliminate extra output that is not of interest in this study. The `VALUE` option is necessary because O'Hara and Shaw use the CRSP value weighted index instead of the Eventus default equally weighted index. The `POST=5` and `PRE=5` specifications limit the event period to days -5 through $+5$. Finally, `TAIL=2` means that significance level symbols will reflect two-tailed tests.

Figure 3.2 shows the statements that the user enters in the SAS editor window. (The statements could also be saved in a plain text program file for batch execution.) The `filename request` statement points to the request file; the quoted string shown as an example is in Windows syntax, but all other aspects of the statement are identical across operating environments. Figure 3.3 displays the request file. Note that it is sorted by `Swarynum` here, but in general it need not be sorted at all.

Figure 3.4 shows the first page of Eventus output. The output will be in the SAS output window (or procedure output file in the case of a batch run.)

Figure 3.3
Request file for the "Too Big to Fail" study.

```
48223   840920   01
41718   840920   02
47079   840920   04
58827   840920   05
47896   840920   06
46842   840920   07
51676   840920   08
48071   840920   09
48354   840920   10
50024   840920   11
52863   840920   12
19043   840920   13
53858   840920   14
53903   840920   15
60839   840920   16
51772   840920   17
59379   840920   18
53209   840920   19
59109   840920   20
53938   840920   21
56805   840920   22
26550   840920   69
```

The first page lists the estimation period, summarizes the results of searching the CRSP file for the sample, explains the statistical significance symbols, and may contain other information depending on the options selected.

In figure 3.5 is a listing of the sample and a detailed report of the number of returns Eventus found for each observation. The firm names listed are current as of the event date, based on the CRSP file name structures. The names are truncated in some cases, but if there is a letter distinguishing a class of common stock, Eventus always includes it at the end of the name. None of the 22 sample firms has classified stock. Eventus prints the untruncated name in the wide output format (`PAGE=WIDE` on the `EVENTUS` statement.) If Eventus drops an observation from the sample, an explanatory message appears under "Reason if no usable returns".

Figure 3.6 presents the Eventus parameter estimate listing. The user can suppress this listing by specifying `NOPLIST` on the `EVENTUS` statement. Eventus prints more statistics, including the first-order autocorrelation of the market model residuals, if the `PAGE=WIDE` option appears on the `EVENTUS`

Figure 3.4
The first page of Eventus output.

```
                O'Hara and Shaw, JF, Dec 1990, pp.1587-1600                1
            Replication of Table III.C:  Insolvent Bank Sample

ESTIMATION PERIOD:  Ends 6 days before the event date;
50 days in length.
TOTAL NUMBER OF EVENTS:          22
EVENTS WITH USEABLE RETURNS:     22
EVENTS DROPPED:                   0



STATISTICAL SIGNIFICANCE LEVELS:  2 tailed
NONPARAMETRIC TEST:  Generalized sign test in all event studies.
For nonparametric tests, significance levels of .10, .05, .01 and .001
are denoted by (, <, <<, <<< or ), >, >>, >>> respectively.  Left
brackets -- (, < -- appear when the ratio of positive to negative
is less than in the parameter estimation period.  Right brackets
mean that the ratio is more positive than in the estimation period.



NOTE:  Useable returns means all nonmissing returns except the
first day after a missing estimation period return.
```

statement.

The event study results appear in figure 3.7. The results are substantially the same as O'Hara and Shaw (1990) report. The authors report average and median abnormal returns on day 0 of 0.42% and −0.24%, while Eventus reports 0.43% and −0.24%. If we use the `OUTSAS=` option to save the event study, then run a program with the `EXTRACT` statement (discussed in Chapter 5) to examine the results, we find that 0.0042 is the truncated value of the mean, 0.0042566. Eventus correctly rounds the result to 0.43% (for reporting only, not computation.) Thus truncation may explain the 0.01% discrepancy in the reported average abnormal return. O'Hara and Shaw report a t statistic of 0.78 and Eventus reports 0.77. O'Hara and Shaw compute the estimation period residual standard deviation by dividing the sum of squared differences by 49, or one less than the number of days in the estimation period. Eventus divides by two less than the number of days, because we estimate two market model parameters. The difference in computation of the

Figure 3.5
The Eventus sample listing and input report.

O'Hara and Shaw, JF, Dec 1990, pp.1587-1600 2
Replication of Table III.C: Insolvent Bank Sample
Results of Daily Stock Returns Input

Swarynum	PERMNO	Name on Event Date	Event Date	Estimation Period Returns <=50	Event Period Returns <=11	Reason if no usable returns
1	48223	MANUFACTURERS HAN	09/20/84	50	11	
2	41718	CHASE MANHATTAN C	09/20/84	50	11	
4	47079	CITICORP	09/20/84	50	11	
5	58827	BANKAMERICA CORP	09/20/84	50	11	
6	47896	CHEMICAL NEW YORK	09/20/84	50	11	
7	46842	IRVING BANK CORP	09/20/84	50	11	
8	51676	CROCKER NATIONAL	09/20/84	50	11	
9	48071	MORGAN J P & CO I	09/20/84	50	11	
10	48354	BANKERS TRUST NY	09/20/84	50	11	
11	50024	WELLS FARGO & CO	09/20/84	50	11	
12	52863	EQUIMARK CORP	09/20/84	50	11	
13	19043	MARINE MIDLAND BK	09/20/84	50	11	
14	53858	FIRST CHICAGO COR	09/20/84	50	11	
15	53903	FIRST PENNSYLVANI	09/20/84	50	11	
16	60839	SECURITY PACIFIC	09/20/84	50	11	
17	51772	BANK OF BOSTON CO	09/20/84	50	11	
18	59379	MELLON NATIONAL C	09/20/84	50	11	
19	53209	FIRST WIS CORP	09/20/84	50	11	
20	59109	REPUBLICBANK CORP	09/20/84	50	11	
21	53938	REPUBLIC NEW YORK	09/20/84	50	11	
22	56805	INTERFIRST CORP	09/20/84	50	11	

Figure 3.6
The Eventus parameter estimate listing.

O'Hara and Shaw, JF, Dec 1990, pp.1587-1600 3
Replication of Table III.C: Insolvent Bank Sample

Parameter Estimates and Estimation Period Statistics

Index Weight=Value

SWARYNUM	PERMNO	Alpha	Beta	Mean Return	Market Model Residuals>0	Residual Standard Deviation
1	48223	0.00102	1.120	0.00285	50.00%	0.02192
2	41718	0.00081	1.095	0.00260	48.00%	0.01216
4	47079	0.00019	1.703	0.00297	46.00%	0.01388
5	58827	-0.00042	1.394	0.00185	52.00%	0.01748
6	47896	0.00190	1.433	0.00424	42.00%	0.01909
7	46842	0.00103	1.052	0.00275	46.00%	0.02000
8	51676	0.00716	0.309	0.00766	44.00%	0.03187
9	48071	0.00089	1.042	0.00259	42.00%	0.01129
10	48354	0.00273	1.089	0.00451	50.00%	0.01107
11	50024	0.00395	0.952	0.00550	52.00%	0.01283
12	52863	-0.00140	1.132	0.00045	50.00%	0.02894
13	19043	0.00321	0.464	0.00396	48.00%	0.01535
14	53858	0.00171	1.519	0.00419	44.00%	0.01643
15	53903	0.00258	0.833	0.00394	50.00%	0.02685
16	60839	0.00196	0.822	0.00330	44.00%	0.01171
17	51772	0.00134	1.007	0.00299	48.00%	0.01261
18	59379	-0.00069	1.125	0.00114	44.00%	0.01719
19	53209	0.00210	0.494	0.00291	48.00%	0.01944
20	59109	0.00073	0.858	0.00213	44.00%	0.01932
21	53938	0.00070	0.830	0.00205	44.00%	0.01494
22	56805	0.00478	-0.261	0.00436	32.00%	0.02361
69	26550	0.00254	0.652	0.00360	44.00%	0.01539
MEAN		0.00176	0.939	0.00330	46.00%	0.01788
MEDIAN		0.00153	1.025	0.00298	46.00%	0.01681

residual standard deviation would tend to make the statistic that Eventus reports smaller, and so may explain the difference.[7]

Eventus reports that 10 of the 22 abnormal returns on day 0 are positive, which is the same 45.5% fraction that the original study reports. The authors provide a significance level for the median abnormal return of 0.516. The article does not seem to define the statistic explicitly. Eventus reports a generalized sign test statistic of -0.04 for day 0. The statistic is based on the normal approximation to the binomial, so it has an approximately standard normal distribution. A table of the standard normal distribution shows that the probability of observing a value of $z \leq -0.04$ is 0.5160. Thus O'Hara and Shaw (1990) appear to use the generalized sign test.

The program does not include a `WINDOWS` statement, so Eventus reports three windows that jointly cover the entire event period. In this study, the authors focus on day 0 because there is no reason for any other day to have a stock price reaction.

When you run Eventus, the SAS log window or log file, depending on the mode of SAS operation, reports the completion of data steps and procedures that Eventus executes internally. Most Eventus users will find that these reports have little meaning for them and they may ignore most of what is in the log. However, it is still a good idea to look briefly through the SAS log for messages that begin with `EVENTUS NOTE`, `EVENTUS WARNING`, or `EVENTUS ERROR`. If you contact us for technical support to help resolve a problem with an Eventus run, please include the entire log window contents, or log file, resulting from a single run.

3.3 Abnormal Returns between Paired Events: The `TWIN` Option

The `TWIN` option provides a means of computing portfolio cumulative average abnormal returns over periods that vary in length from one firm to another. The Eventus statements in figure 3.8 run a `TWIN` event study. For the most part, the options also are options for single event date event studies and function similarly in the two contexts. The only differences are the word `TWIN`

[7] If O'Hara and Shaw used the same untruncated average abnormal return that Eventus used, their result would differ from ours by a factor of $\sqrt{(49/48)} = 1.010363$; and $1.010363 \times 0.77 = 0.78$, rounded.

Figure 3.7
Event study results.

O'Hara and Shaw, JF, Dec 1990, pp.1587-1600 4
Replication of Table III.C: Insolvent Bank Sample

Market Model, VW Index

Day	Average Abnormal Return	Median Abnormal Return	t	N	Positive: Negative	Generalized Sign Z
-5	1.27%	1.51%	2.29*	22	16:6	2.52>
-4	1.21%	1.08%	2.18*	22	17:5	2.94>>
-3	0.27%	0.14%	0.49	22	12:10	0.80
-2	0.18%	0.32%	0.33	22	15:7	2.09>
-1	0.21%	-0.12%	0.38	22	10:12	-0.05
0	0.43%	-0.24%	0.77	22	10:12	-0.05
+1	0.30%	0.29%	0.54	22	15:7	2.09>
+2	-0.55%	-0.84%	-1.00	22	8:14	-0.91
+3	-0.38%	-0.41%	-0.68	22	9:13	-0.48
+4	0.61%	-0.46%	1.10	22	8:14	-0.91
+5	0.33%	-0.12%	0.59	22	10:12	-0.05

Days	Cumulative Average Abnormal Return	Median Cumulative Abnormal Return	t	Positive: Negative	Gen Sign Z
(-5,-2)	2.92%	3.40%	2.64**	19:3	3.81>>>
(-1,0)	0.63%	0.44%	0.81	12:10	0.81
(+1,+5)	0.30%	-1.14%	0.24	7:15	-1.33

$, (,) significant at .10 *, <, > significant at .05
, <<, >> significant at .01 *, <<<, >>> significant at .001

Figure 3.8
Eventus statements for event studies cumulating returns between two firm-specific event dates.

filename request 'G:\Some Folder\Filename.extension';

EVENTUS TWIN [**LIBNAME=CRSP**] [**MONTHLY**|**EXCESS**]
[**ESTINTER=DAY**|**MONTH**] [**PAGE=WIDE**];

[**TITLE** *'text'*;]

[**TITLE2** *'text'*;]

REQUEST [**CUSIP**|**CUSIPERM**] [**ID=***variable* **IDFMT=***format*]
[**GROUP=***variable*] [**GRWEIGHT**]
[**DATEFMT=MMDDYY**|**YYMMDD**|**DDMMYY**|**DATE**|**CRSP**]
[**AUTODATE**] [**NODIVIDX**] [**SP500**|**COMPOSIT**]
[**EST=**−*value*|+*value*] [**POOL**] [**ESTLEN=***n*]
[**MINESTN=***n*] [**SHORT**] ;

WINDOWS [**EVENT1=***descriptor1* **EVENT2=***descriptor2*];

EVTSTUDY [**NONAMES**|**NOPLIST**|**SIC**] [**DETAIL**]
[**NOSTD**] [**VALUE**|**BOTH**|**SPORT**] [**SW**|**GARCH**|**EGARCH**]
[**CSECTERR**] [**STDCSECT**|**EGLS**|**CDCSI**] [**SERIAL**]
[**MEDIAN**] [**BUYHOLD**] [**TAIL=1**|**2**]
[**OUTSAS=***libref.membername* **PACKAGE=***specifier]*;

on the `EVENTUS` statement and the specification of the `WINDOWS` statement. The "descriptors" in the `WINDOWS` statement are 1–11 character names by which you want Eventus to designate the two event dates on the output. Blank spaces are not allowed within either descriptor. See Section 3.1 for descriptions of the remaining options.

Each line of the request file must contain a pair of event dates, separated by one or more blank spaces, in the location where the single event date goes in a normal event study request file.

Figure 3.9
Eventus statements for reprinting and merging saved event studies.

EVENTUS [**TWIN**] [**MONTHLY**|**EXCESS**] [**PAGE=WIDE**];

[**TITLE** *'text'*;]

[**TITLE2** *'text'*;]

WINDOWS [*(begin,end) (begin,end) ...*];

OLDSTUDY **INSAS=***libref.membername*
[**INSAS2=***libref.membername*]
[**INSAS3=***libref.membername*]
[**ID=***ID_variable_name* **IDFMT=***format*]
[**DETAIL**|**DETAIL=FULL**] [**NOSTD**|**STDONLY**]
[**RANKTEST**] [**SERIAL**] [**TAIL=1**|**2**] ;

3.4 Reprinting or Merging Saved Event Studies with the OLDSTUDY Statement

The EVTSTUDY statement allows you to save abnormal returns in a SAS data set with the OUTSAS option. The OLDSTUDY statement lets you reprint a single event study (possibly with a different set of windows) or merge two or three event studies from a saved EVTSTUDY SAS data set.

Figure 3.9 lists the statements and options used in an OLDSTUDY program. Most of the options can be used in EVTSTUDY programs, and are described in Section 3.1 above. The options have a similar meaning here, but placement and usage differ in certain instances. For example, the ID= option goes on the OLDSTUDY statement and specifies the name of the original identification variable.

The TWIN, MONTHLY, WEEKLY or EXCESS option is needed on the EVENTUS statement, when used in the program that created the saved event study data, only for proper labeling of OLDSTUDY output. No other option is needed on the EVENTUS statement. The windows specified on the WINDOWS statement need not be those used with the original EVTSTUDY program(s).

The INSAS specification and the INSAS2 and INSAS3 options on the OLDSTUDY statement tell Eventus where to find saved event study SAS data sets. The *libref* and *membername* parameters should match those on the OUTSAS= options of the original EVTSTUDY statement(s). (See page 31.) The event studies to be merged all should have had the same PRE, POST and abnormal return method options in the original programs. If the original programs specified different variable names for ID=, or incompatible identifying variable formats (such as numeric and character), omit the ID= option from the OLDSTUDY statement.

Selecting benchmarks and standardized tests

In general, the OLDSTUDY statement detects the abnormal return methods used in the original event study and takes them as the defaults. To omit market adjusted returns, specify NOMAR. To omit market model returns, specify NOMM. To omit just the standardized abnormal return tests using the market model, specify NOSTD. To omit everything *except* the market model with standardized abnormal return tests, specify STDONLY.

To tell Eventus to report standardized abnormal return tests for non-market model methods, specify STDALL on the OLDSTUDY statement.

See Section 3.1 for a discussion of the DETAIL, RANKTEST, SERIAL and TAIL options.

Figure 3.10 displays a sample Eventus program to merge two event studies. Notice that because Eventus does not need to read a request file or a CRSP database, you do not use a REQUEST statement with OLDSTUDY.

Figure 3.10
Sample Eventus program using OLDSTUDY to merge two saved event studies.

If you create and save data sets with the following programs...

```
EVENTUS;                          |  EVENTUS;
 REQUEST ...;                     |   REQUEST ...;
 EVTSTUDY OUTSAS=MYLIB.RES1;      |   EVTSTUDY OUTSAS=YOURSAS.RES2;
```

...you can merge the two portfolios and print the combined event study with a third:

```
EVENTUS;
 TITLE  'text';
 TITLE2 'text';
 WINDOWS (-45,-2) (+1,+5);
 OLDSTUDY INSAS=MYLIB.RES1 INSAS2=YOURSAS.RES2 TAIL=2;
```

Chapter 4

Event Studies Using Non-CRSP Data

While the CRSP database is the only data source from which Eventus automatically retrieves and assembles data, you can run an Eventus event study with other data too. This chapter describes how to conduct an event study where Eventus reads stock return data that the researcher has already retrieved and organized. If your data source provides only price and index level data, you will need to calculate returns before running Eventus.

Datastream users can automate most of the process using the EventStream package from Cowan Research, L.C. EventStream accepts Eventus-like options and request files, generates ready-to-run DSWindows macros for use with the *Datastream* terminal application, pre-processes downloaded data, builds `USERSTOK` files, and generates Eventus statements. Please see our `www.eventstudy.com` www.eventstudy.com web site for details.

4.1 Event Studies Centered on a Single Event Date

Figure 4.1 displays the Eventus statements to run a single-event date event study. The options are described below.

Figure 4.1

Eventus statements for an event study centered around a single date.

filename request 'G:\Some Folder\Filename.extension';
filename userstok 'G:\Some Folder\Filename.extension';

EVENTUS NONCRSP [**MONTHLY**|**WEEKLY**|**QUARTERLY**]
[**PAGE=WIDE**];

[**TITLE** *'text'*;]

[**TITLE2** *'text'*;]

REQUEST [**ID=***variable* **IDFMT=***format*]
[**DATEFMT=MMDDYY**|**YYMMDD**|**DDMMYY**|**DATE**]
[**EST=**−*value*| +*value*] [**POOL**] [**ESTLEN=***n*] [**NAME**] ;

WINDOWS [*(begin,end) (begin,end) ...*];

EVTSTUDY [**NOPLIST**] [**DETAIL**|**DETAIL=FULL**]
[**RAW**] [**CP**] [**NOMAR**] [**NOMM**|**NOSTD**|**STDONLY**|**STDALL**]
[**VALUE**|**BOTH**] [**SW**|**GARCH**|**EGARCH**]
[**PRE=***periods*] [**POST=***periods*] [**OVERLAP**]
[**CSECTERR**] [**STDCSECT**] [**SERIAL**]
[**RANKTEST**] [**MEDIAN**] [**BUYHOLD**] [**TAIL=1**|**2**]
[**OUTSAS=***libref.membername*]
[**PACKAGE=***specification*] ;

The EVENTUS statement

The NONCRSP option on the EVENTUS statement indicates that you are supplying the stock and index return data for the event study, so that Eventus does not expect to search a CRSP database. A SAS `filename userstok` statement or the operating system must associate the fileref USERSTOK with the file containing the returns. The content and format of this file is discussed under the REQUEST statement below.

MONTHLY|WEEKLY|QUARTERLY With non-CRSP returns input, the return interval options provide only a description of the data for labeling purposes; they have no effect on the analysis. Eventus uses the description to label the results, but otherwise the options have no effect. (You also can specify ANNUAL for annual data.)

PAGE=WIDE Normally Eventus prints portrait-oriented, or tall, pages. The default output is easy to view on a display screen before printing, and is suitable for almost any printer. The PAGE=WIDE option on the EVENTUS statement allows Eventus to print landscape-oriented, or wide, output. With wide output, more information will fit on one page.[1] The disadvantage of wide output is that it produces a line width of 132 characters. To print wide output, you need a printer capable of landscape orientation or a small font.

The REQUEST statement

The REQUEST statement instructs Eventus to read the request file and the user-supplied returns file. The request file is an external file; you use it to supply the identifiers for your sample, the event dates, and optional other information. Eventus does not compare the identifiers and event dates in the request file to any external database. The identifier is referred to as CUSIPs in this chapter, but it need not be a true CUSIP; it can be any one to eight character alphanumeric value, padded by blanks if it is shorter than eight characters. The date must be a valid calendar date but is not subject to any other limitation.

Each line of the request file contains data for a single security-event date combination. At a minimum, each line must contain a CUSIP and event date.

[1] The ALLDAYS option (see page 51 below) is available with either wide or tall output to force Eventus to use additional pages to avoid the limitation on the number of days.

For example, a line of the request file could look like this: 80234H10 950920 Spacing does not matter as long as the CUSIP comes first, then the event date, with at least one space between.

The rest of this section describes `REQUEST` statement options for the processing of dates and market and stock returns and for the construction of the estimation period. A SAS `filename` statement or the operating system must associate the fileref `REQUEST` with the request file.

Including an identification variable

Each line of your request file may include an identification variable following the event date. Eventus will read it, and print it on subsequent output, if you name it with `ID=`. `ID=ID` works, or you can choose another name. Use `IDFMT` to tell Eventus what format to use for reading and printing the identifying variable. `IDFMT=4` means a 1–4 digit integer, while `IDFMT=$4` means a four-character string. Other lengths and other SAS formats are permitted also. With `IDFMT=$4` specified, a line of your request file might look like this:

```
16372210 19840920 AY4E
```

Using date formats other than YYMMDD

The `REQUEST` statement allows you to specify the format of dates in your request file. You may list calendar dates in nearly any conventional format. If you use either six- or eight-digit YYMMDD, the default, you need not specify `DATEFMT`. Besides MMDDYY and DDMMYY, you can use the SAS date format DATE. (An example of a date in DATE format is 19OCT1987.)

Options for constructing the estimation period

The estimation period is the time period used for running market model regressions, computing comparison period mean returns, and so on.

`EST=` and `POOL` By default, Eventus determines the estimation period for each firm by subtracting 46 trading days from the event date in your request file. The resulting date becomes the *last* day of the estimation period. If you want the estimation period moved back, say to 90 days before the event date, specify `EST=-90`. If you need an estimation period following the event

date, specify a positive number. `EST=+61` (the plus sign is optional) gets you an estimation period that *begins* on day +61.

The default estimation period for weekly event studies ends with week −10; for monthly event studies, the default is to end the estimation period with month −13.

`EST` has no effect on the length of the estimation period. The `ESTLEN` option, described next, changes the estimation period length.

`ESTLEN=` By default, the estimation period is 255 days long when using daily data and 52 weeks or 60 months long when using weekly or monthly data. You can change the estimation period length with `ESTLEN`. The largest number you can use is 999 and the smallest is 3. Eventus assumes that you are specifying the number of months for a monthly event study, weeks for a weekly event study, or days for a daily event study.

Reading firm names from the request file

To have Eventus print security issuer names as it does when using CRSP database data, include the name as the last item on each line of the request file and specify the `NAME` option on the `REQUEST` statement. Up to 33 characters will be used.

How to construct the `USERSTOK` return data file

To manually build a `USERSTOK` file, place security and index return data in an external file, using one line for each trading day (or month or other period) for each security. Each line in the request file must have a corresponding set of lines in the `USERSTOK` external file. The number of lines for each security must be the same. For example, with the default 255-day estimation period and 61-day event period, each line in the request file must correspond to 316 consecutive lines in the `USERSTOK` file. On each line, include the following information: CUSIP, identifying variable if you specified `ID=`, rate of return for the security, rate of return for the market index. If you specified `BOTH` on the `EVTSTUDY` statement (see below), list the equally weighted index return first, then the value-weighted index return.

When a security or index return is missing, use the SAS missing value code, a decimal point. Be sure to leave at least one blank space on each side of the missing value code.

In many situations, an easier approach is to use SAS data sets to input the USERSTOK data. This chapter does not provide the details of the approach, but reference documentation for the INSAS option used with the SAS data set approach appears on page 131. Chapter 6 presents an example of SAS data set input that the user may adapt as needed. Additional examples appear on our web site.

The WINDOWS statement

The WINDOWS statement is optional. Use it to specify ranges of dates — "windows" — in event time over which Eventus should report cumulative or compounded abnormal returns and associated test statistics. Each window specification requires a *begin* and *end*, which can be the same if desired. For example, (−3,0) specifies a four day or four month window ending with the event date; (2,2) specifies a one day or one month window containing the second day or month after the event date. When *begin* and *end* are equal, no cumulation or compounding occurs; the only purpose for such a specification would be to cause a single date to appear in the window output section for ease of reference. If the WINDOWS statement is omitted, Eventus reports three windows: (−PRE,−2), (−1,0) and (+1,+POST). The results for the windows appear at the bottom of the same page on which the results for individual dates are printed, or on a separate page if there is not enough space.

The EVTSTUDY statement

The EVTSTUDY statement tells Eventus to run an event study. The options permit you to change the default event period, select market indices and methods for computing abnormal returns and test statistics, increase or decrease the amount of printed output, and store results in a permanent SAS data set.

Selecting the amount of printed output

EVTSTUDY produces a report of the results of reading your data. The report lists the CUSIP, identifying variable and event date from your request file, and the name of the issuer if you specified the NAME option on the REQUEST statement. The report also tells, for each firm, how many returns Eventus

found in the estimation period and in the event period. If you prefer not to get this report, specify the NONAMES option.

Unless you specify NOPLIST on the EVTSTUDY statement, you also get a list of the estimation period return statistics for each observation and the mean and median for the sample. The report includes market model alphas, betas and residual standard deviations, fraction of market model residuals positive, and additional statistics if space permits.

The last set of output from EVTSTUDY contains the event time results.

By default, one page is generated for each abnormal return method. Each page lists the portfolio mean abnormal return, test statistic, number positive and negative, and significance levels, for the event date ("day 0") and as many as 34 surrounding days (94 if PAGE=WIDE appears on the EVENTUS statement), depending on PRE and POST.[2] At the bottom of the page, for each window you request, are the portfolio cumulative abnormal return, the test statistics, the number positive and negative and significance symbols.

You may want to see window or daily abnormal returns for individual firms. EVTSTUDY will print individual firm abnormal returns only for the standardized abnormal return method. To obtain window cumulative abnormal returns for every firm, specify DETAIL on the EVTSTUDY statement. As many firms will appear on a single page as there is room for. To obtain complete daily abnormal returns for each firm, specify DETAIL=FULL. Full detail takes at least one page for each firm, so be prepared for lots of output. If you need to look at only a few individual days, use the following trick. Suppose you want to see day +1 for every firm. On the WINDOWS statement, include (1,1). Then specify DETAIL (*not* DETAIL=FULL) on the EVTSTUDY statement. This way you get the day you need without all the extra output. DETAIL specifications are ignored if you specify NOMM or NOSTD unless you also specify STDALL.

For individual raw or abnormal returns from non-standardized method event studies, see the EXTRACT statement in Chapter 5.

Specifying benchmarks and standardized or non-standardized tests

The default is to compute three sets of abnormal returns and test statistics: market adjusted returns, market model abnormal returns using portfolio

[2]To print all the dates in the range defined by PRE and POST when the number exceeds 35 (or 95 with PAGE=WIDE), see the ALLDAYS option below (page 51.)

time-series standard errors and market model abnormal returns using standardized abnormal return tests. To add comparison period mean adjusted returns, specify CP on the EVTSTUDY statement. To add total unadjusted, or "raw", returns, specify RAW. To omit market adjusted returns, specify NOMAR. To omit market model returns altogether, specify NOMM. To omit just the standardized abnormal return tests using the market model, specify NOSTD. To omit everything *except* the market model with standardized abnormal return tests, specify STDONLY.

These options can be combined. For example, CP NOMAR NOMM causes EVTSTUDY to analyze comparison period mean adjusted returns, and omit all other types of abnormal returns.

To tell Eventus to report standardized abnormal return tests for non-market model methods, specify STDALL on the EVTSTUDY statement.

Selecting the value weighted market index

Normally, Eventus labels the index return as equally weighted. Specify VALUE to label the index as value weighted index or BOTH if you supplied two index return series and want results for each.

Scholes-Williams and GARCH market model estimation

Eventus reports market model results using both ordinary least squares and Scholes-Williams (1977) beta estimation when you specify SW.

When the GARCH option is specified, Eventus estimates the market model assuming a GARCH(1,1) error structure. For exponential GARCH, or EGARCH(1,1), specify the EGARCH option. With either the GARCH or EGARCH option, maximum likelihood estimates using the quasi-Newton algorithm are produced. Eventus reports the alpha and beta estimates as it does in the case of ordinary least squares, but does not report the estimated parameters of the conditional error variance model. No more than 40 iterations will be performed for each stock. In general, convergence will be better the longer the estimation period (ESTLEN) and better with the EGARCH(1,1) than the GARCH(1,1) model.

Cowan and Sergeant (1996) report that event study test specification and power are insensitive to the use of Scholes-Williams versus OLS estimation. Corhay and Rad (1996) and Brockett, Chen and Garven (1999) discuss the potential benefits of GARCH estimation in event studies. Bollerslev, Chou and

Kroner (1992) provide an overview of GARCH, EGARCH and related models in finance.

Specifying the number of days or months in the event period

EVTSTUDY computes and reports abnormal returns for the event period, which is defined by default as days or weeks −30 through +30, or months −12 through +12. To change from these defaults, specify the number of days before and after on the EVTSTUDY statement, using PRE and POST. These options may be used singly or in combination. For example, PRE=60 means that the abnormal returns are to start with day −60; unless POST= is also specified, the latter retains its default value.

If the event period length (PRE+POST+1) exceeds 35 days, weeks or months, only 35 normally are printed (95 if the PAGE=WIDE option is in effect.) Any windows specified on a WINDOWS statement are printed, however, regardless of the range of dates covered. If you want to print results for all the dates individually, specify ALLDAYS on the EVTSTUDY statement. For further discussion, see page 130 in Appendix B.

If you specify PRE= or POST=, you may need to change the estimation period from the default. To change the estimation period, specify EST= on the REQUEST statement. Eventus will stop and print an error message if the event period defined by PRE and POST overlaps the default or specified estimation period. If for some reason you want to override overlap checking and allow the estimation period and the event period to have dates in common, specify OVERLAP on the EVTSTUDY statement.

Computing cross-sectional standard errors instead of time series standard errors

Normally Eventus computes portfolio standard errors from the time series of estimation period portfolio abnormal returns. The CSECTERR option specifies that the standard error for each date in event time should be computed across securities instead. For an example of the application of the cross-sectional method, see Pilotte (1992). This option has no effect on the standardized abnormal return method.

The standardized cross-sectional test

The z statistic that Eventus normally computes for the standardized abnormal return method is the widely used statistic described by Patell (1976). The STDCSECT option substitutes the standardized cross-sectional test. This is an extension of the Patell test introduced by Boehmer, Musumeci and Poulsen (1991). The standardized cross-sectional test compensates for a possible variance increase on an event date by performing a simple cross-sectional variance adjustment. The SERIAL option explained below is automatically invoked by the STDCSECT option.

EGLS and Collins-Dent tests

The EGLS and CDCSI options replace the default standardized test with the estimated generalized least squares test and Collins-Dent test assuming cross-sectional independence, respectively. These tests are discussed in detail by Sanders and Robins (1991). The EGLS and CDCSI options automatically invoke the SERIAL option.

Adjusting window z tests for serial dependence

The SERIAL option applies to standardized abnormal returns only. Normally, the test statistics for abnormal returns cumulated over intervals you define in the WINDOWS statement are not adjusted for serial dependence. Mikkelson and Partch (1988) and others perform such a correction on cumulative returns. The SERIAL option causes Eventus to apply the correction to standardized abnormal returns only. Karafiath and Spencer (1991) and Cowan (1993) provide simulation evidence of the properties of the corrected and uncorrected test statistics.

Note that the serial dependence that the SERIAL option corrects is not due to any presumed dependence in the true market model error term, but occurs because all of the abnormal return estimators being cumulated are functions of the same estimators of the market model parameters. The derivation of the corrected standard error used by Mikkelson and Partch (1988) requires that the abnormal return be interpreted as a forecast error.

Alternative nonparametric tests

Normally, Eventus prints the generalized sign test statistic (see Cowan, 1992) for each day in the event period and for each window with both standardized and non-standardized tests. If you specify `RANKTEST` on the `EVTSTUDY` statement, Eventus instead will report the nonparametric rank test introduced by Corrado (1989) with non-standardized tests. The `JACKNIFE` option produces the jackknife test developed by Giaccotto and Sfiridis (1996). The `RANKTEST` and `JACKNIFE` tests are mutually exclusive. The generalized sign test still will be reported with standardized abnormal return method results.

Reporting median abnormal returns

This option is needed only when the `PAGE=WIDE` option appears on the `EVENTUS` statement. With the default output format, `PAGE=TALL`, Eventus always reports median and mean abnormal returns. With the optional wide output format, Eventus reports median abnormal returns in place of the number positive and negative if you specify `MEDIAN` on the `EVTSTUDY` statement. Significance symbols for the default generalized sign test or optional rank test still are reported.

Computing buy-and-hold compounded window returns

To obtain buy-and-hold abnormal returns for multiperiod windows, specify `BUYHOLD` on the `EVTSTUDY` statement. The option applies only to non-standardized abnormal return methods. (By default, Eventus reports non-standardized results for each abnormal return method used in an event study, and standardized results in addition for the market model abnormal returns.) Eventus computes buy-and-hold abnormal returns by compounding successive daily (or other period) raw returns and market index returns, then adjusting the raw returns according to the abnormal return method used. Comparison period mean returns and market model alphas are adjusted for the window length. The Eventus output labels the mean buy-and-hold window returns "Average Compounded Abnormal Return," while the default additive window abnormal returns appear as "Cumulative Average Abnormal Return."

Reporting one- instead of two-tailed tests

Eventus reports all significance levels for one-tailed tests by default. `TAIL=2` changes to two-tailed tests.

Saving results to a file

Normally, none of the event study results are saved in a disk file. To store firm-by-firm results, specify the SAS file to store them in using the `OUTSAS=` option.

The *libref* parameter specifies the libref of the SAS data library in which to store the results. Eventus will create a new data set unless the one named already exists in the SAS data library, in which case the existing one will be overwritten. A SAS `libname` statement or the operating system must associate the libref with the aggregate storage location (a folder or directory on most systems) used as the library.

A related option, `PACKAGE=`, determines the content of the file. If you intend to use `OLDSTUDY` to reprint or merge event study results, specify `PACKAGE=1`. If you plan to use `EXTRACT` to create dependent variables for cross-sectional analysis of abnormal returns (see Chapter 5), specify `PACKAGE=DG`, or just `PACKAGE=D` if you do not need weights or standardized returns. Table B.1 on page 134 lists the complete set of `PACKAGE` specifications available.

4.2 An Event Study Example

In this section we replicate part of the O'Hara and Shaw (1990) study as in Chapter 3 using non-CRSP return data. Figure 4.2 shows the Eventus program to run the replication. The `EVENTUS` statement needs only the `NONCRSP` option. O'Hara and Shaw state that their estimation period is days −55 through −6, relative to the event date of September 20, 1984. To replicate their estimation period we specify `EST=-6` and `ESTLEN=50` on the `REQUEST` statement. This means that the first 50 days of returns data for each firm must correspond to this estimation period. We identify the individual bank stocks using an abbreviated firm name, calling the identifying variable `Bankname`. The specification `IDFMT=$10.` means that the identifying values are up to ten characters long. No `ID` variable is required; the CUSIP alone is enough. If there is no separate identifying variable, `ID` and `IDFMT` are omitted.

O'Hara and Shaw (1990) report only market model abnormal returns; the option NOMAR on the EVTSTUDY statement suppresses the computation and reporting of market adjusted returns. Because the firms are all in the same industry and have a common event date, the authors do not use a standardized residual test, but base their test statistic "on a standard deviation estimated for the portfolio of sample firms from residual returns in the estimation period." [See Chandra and Balachandran (1990) and Chandra, Moriarity and Willinger (1990) for further discussion of cross-sectional dependencies in event studies.] We suppress standardized tests with the NOSTD option. The NOMAR and NOSTD options are not necessary; they only eliminate extra output that is not of interest in this study.

Because no WINDOWS statement is specified, Eventus automatically generates multiday windows around the event date to supplement the day-by-day results. The BUYHOLD option on the EVTSTUDY statement produces buy-and-hold results for the windows around the event date. The default when the user does not specify BUYHOLD is to cumulate (additively) abnormal returns across windows.

The VALUE option serves only to label the Eventus results as using a value-weighted index. O'Hara and Shaw use a value weighted index, for which we have collected data. The POST=5 and PRE=5 specifications mean that the event period returns span days -5 through $+5$. Thus, after the 50 days of estimation period data for each firm, we must provide 11 days of event period data centered on September 20, 1984. TAIL=2 means that significance level symbols will reflect two-tailed tests. Finally, SKIP=1 instructs Eventus to ignore the first line of the USERSTOK returns file.

Figure 4.2
Eventus program for an event study of the "Too Big to Fail" policy.

```
filename request 'C:\Folder\request_filename.ext';
filename userstok 'C:\Folder\userstok_filename.ext';
eventus noncrsp;
title ''O'Hara and Shaw, JF, Dec 1990, pp.1587-1600'';
title2 ''Replication of Table III.C:  Insolvent Bank Sample'';
request est=-6 estlen=50 id=Bankname idfmt=$10.;
evtstudy nomar nostd buyhold value post=5 pre=5 tail=2 skip=1;
```

Figure 4.3
Part of the request file for the "Too Big to Fail" study.

```
56480910    840920  Manufactur
16161010    840920  ChaseManha
17303410    840920  Citicorp
06605010    840920  Bankameric
```

Figure 4.3 displays the first few lines of the request file. Note that Eventus automatically sorts the request file. The CUSIPs could be any eight-character strings as long as they are consistent between the request file and the returns file.

Figure 4.4 shows the first 62 lines of the stock and index return file associated with the fileref `USERSTOK`. The first line is a header line for the user's reference; Eventus ignores this line because of the `SKIP` option on the `EVTSTUDY` statement. The next 61 lines contain the daily data for the 50-day estimation period followed by the 11-day event period. The file contains $1 + 61 \times 22 = 1343$ lines in all. Eventus reads only the first four columns of data from this file. We happen to have included the calendar date on each line for reference, but Eventus does not use this information in any way. Had we not specified the `ID` option on the `REQUEST` statement, we would have needed only three columns in this file, for the CUSIP, stock return, and index return.

The first page of Eventus output lists the estimation period, describes the sample size, explains the statistical significance symbols, and may contain other information depending on the options selected. For an example, see Chapter 3.

In figure 4.5 is a listing of the sample and a detailed report of the number of returns Eventus found for each observation.

Figure 4.6 presents the Eventus parameter estimate listing. The user can suppress this listing by specifying `NOPLIST` on the `EVENTUS` statement. Eventus prints more statistics, including the first-order autocorrelation of the market model residuals, if the `PAGE=WIDE` option appears on the `EVENTUS` statement.

The event study results appear in figure 4.7. The results are substantially the same as O'Hara and Shaw (1990) report. Chapter 3 has a detailed discussion.

Figure 4.4
First 62 lines of the user-supplied returns file for the "Too Big to Fail" study.

CUSIP	Bank name	RETURN	INDEX	Date
56480910	Manufactur	-0.010152	0.003478	840703
56480910	Manufactur	-0.020513	-0.005392	840705
56480910	Manufactur	-0.041885	-0.003201	840706
56480910	Manufactur	0.043716	0.006032	840709
56480910	Manufactur	0.005236	-0.002348	840710
56480910	Manufactur	0.015625	-0.013716	840711
56480910	Manufactur	0.020513	-0.003855	840712
56480910	Manufactur	0.035176	0.005025	840713
56480910	Manufactur	0.004854	0.003065	840716
56480910	Manufactur	0.038647	0.004453	840717
56480910	Manufactur	-0.009302	-0.005560	840718
56480910	Manufactur	-0.014085	-0.006433	840719
56480910	Manufactur	-0.038095	-0.005262	840720
56480910	Manufactur	0.004950	-0.005737	840723
56480910	Manufactur	-0.039409	-0.006702	840724
56480910	Manufactur	0.010256	0.005587	840725
56480910	Manufactur	0.015228	0.008092	840726
56480910	Manufactur	0.035000	0.007644	840727
56480910	Manufactur	0.009662	-0.004767	840730
56480910	Manufactur	0.000000	0.002720	840731
56480910	Manufactur	0.004785	0.021275	840801
56480910	Manufactur	0.009524	0.024754	840802
56480910	Manufactur	0.018868	0.028282	840803
56480910	Manufactur	0.050926	0.002992	840806
56480910	Manufactur	-0.017621	0.000459	840807
56480910	Manufactur	-0.008969	-0.004584	840808
56480910	Manufactur	0.027149	0.021686	840809
56480910	Manufactur	0.008811	0.001881	840810
56480910	Manufactur	-0.008734	-0.000570	840813
56480910	Manufactur	-0.079295	-0.004966	840814
56480910	Manufactur	0.000000	-0.008240	840815
56480910	Manufactur	0.023923	0.005825	840816
56480910	Manufactur	0.000000	0.001539	840817
56480910	Manufactur	-0.004673	0.003809	840820
56480910	Manufactur	0.014085	0.016093	840821
56480910	Manufactur	0.009259	-0.003148	840822
56480910	Manufactur	-0.009174	0.001218	840823
56480910	Manufactur	0.000000	0.002024	840824
56480910	Manufactur	-0.013889	-0.005190	840827
56480910	Manufactur	0.000000	0.005274	840828
56480910	Manufactur	0.018779	-0.000454	840829
56480910	Manufactur	0.004608	-0.002779	840830
56480910	Manufactur	-0.004587	0.001160	840831
56480910	Manufactur	-0.013825	-0.008776	840904
56480910	Manufactur	-0.037383	-0.003732	840905
56480910	Manufactur	0.019417	0.007350	840906
56480910	Manufactur	0.014286	-0.006375	840907
56480910	Manufactur	0.009390	-0.000346	840910
56480910	Manufactur	0.027907	0.001752	840911
56480910	Manufactur	0.013575	0.000353	840912
56480910	Manufactur	0.080357	0.017459	840913
56480910	Manufactur	0.024793	0.006062	840914
56480910	Manufactur	0.004032	0.000738	840917
56480910	Manufactur	-0.004016	-0.005255	840918
56480910	Manufactur	-0.016129	-0.003625	840919
56480910	Manufactur	0.028689	0.003045	840920
56480910	Manufactur	-0.007968	-0.008275	840921
56480910	Manufactur	-0.016064	-0.002395	840924
56480910	Manufactur	0.004082	0.001179	840925
56480910	Manufactur	0.028455	0.003405	840926
56480910	Manufactur	0.005217	0.003868	840927

Figure 4.5
The Eventus sample listing and input report.

O'Hara and Shaw, JF, Dec 1990, pp.1587-1600 2
Replication of Table III.C: Insolvent Bank Sample

Results of Daily Stock Returns Input

Bankname	CUSIP	Event Date	Esti-mation Period Returns <=50	Event Period Returns <=11
Bankameric	06605010	09/20/84	50	11
BankersTru	06636510	09/20/84	50	11
BankofBost	06071610	09/20/84	50	11
ChaseManha	16161010	09/20/84	50	11
ChemicalNY	16372210	09/20/84	50	11
Citicorp	17303410	09/20/84	50	11
CrockerNat	22682210	09/20/84	50	11
Equimark	29443250	09/20/84	50	11
FirstChica	31945510	09/20/84	50	11
FirstInter	32054810	09/20/84	50	11
FirstPenn	33607210	09/20/84	50	11
FirstWisco	33761C10	09/20/84	50	11
Interfirst	45891610	09/20/84	50	11
Irving	46371210	09/20/84	50	11
JPMorgan	61688010	09/20/84	50	11
Manufactur	56480910	09/20/84	50	11
MarineMidl	56828710	09/20/84	50	11
MellonNati	58550910	09/20/84	50	11
Republicba	33616010	09/20/84	50	11
RepublicNY	76071910	09/20/84	50	11
SecurityPa	81482310	09/20/84	50	11
WellsFargo	94974010	09/20/84	50	11

Figure 4.6
The Eventus parameter estimate listing.

O'Hara and Shaw, JF, Dec 1990, pp.1587-1600 3
Replication of Table III.C: Insolvent Bank Sample
Parameter Estimates and Estimation Period Statistics

---------- Index Weight=Value ----------

BANKNAME	CUSIP	Alpha	Beta	Mean Return	Market Model Residuals>0	Residual Standard Deviation
Bankameric	06605010	-0.00042	1.394	0.00185	52.00%	0.01748
BankersTru	06636510	0.00273	1.089	0.00451	50.00%	0.01107
BankofBost	06071610	0.00134	1.007	0.00299	48.00%	0.01261
[lines omitted]						
ChaseManha	16161010	0.00081	1.095	0.00260	48.00%	0.01216
WellsFargo	94974010	0.00395	0.952	0.00550	52.00%	0.01283
	MEAN	0.00176	0.939	0.00330	46.00%	0.01788
	MEDIAN	0.00153	1.025	0.00298	46.00%	0.01681

When you run Eventus, the SAS log window or log file, depending on the mode of SAS operation, reports the completion of data steps and procedures that Eventus executes internally. Most Eventus users will find that these reports have little meaning for them and they may ignore most of what is in the log. However, it is still a good idea to look briefly through the SAS log for messages that begin with `EVENTUS NOTE`, `EVENTUS WARNING`, or `EVENTUS ERROR`. If you contact us for technical support to help resolve a problem with an Eventus run, please include the entire log window contents, or log file, resulting from a single run.

4.3 Abnormal Returns between Paired Events: The `TWIN` Option

The `TWIN` option provides a means of computing portfolio cumulative average abnormal returns over periods that vary in length from one firm to another. The Eventus statements in figure 4.8 run a `TWIN` event study. For the most part, the options also are options for single event date event studies and function similarly in the two contexts. The only differences are the word `TWIN`

Figure 4.7
Event study results.

O'Hara and Shaw, JF, Dec 1990, pp.1587-1600 4
Replication of Table III.C: Insolvent Bank Sample

Market Model, VW Index

Day	Average Abnormal Return	Median Abnormal Return	t	N	Positive: Negative	Generalized Sign Z
-5	1.27%	1.51%	2.29*	22	16:6	2.52>
-4	1.21%	1.08%	2.18*	22	17:5	2.94>>
-3	0.27%	0.14%	0.49	22	12:10	0.80
-2	0.18%	0.32%	0.33	22	15:7	2.09>
-1	0.21%	-0.12%	0.38	22	10:12	-0.05
0	0.43%	-0.24%	0.77	22	10:12	-0.05
+1	0.30%	0.29%	0.54	22	15:7	2.09>
+2	-0.55%	-0.84%	-1.00	22	8:14	-0.91
+3	-0.38%	-0.41%	-0.68	22	9:13	-0.48
+4	0.61%	-0.46%	1.10	22	8:14	-0.91
+5	0.33%	-0.12%	0.59	22	10:12	-0.05

Days	Average Compounded Abnormal Return	Median Compounded Abnormal Return	t	Positive: Negative	Gen Sign Z
(-5,-2)	2.96%	3.48%	2.68**	19:3	3.80>>>
(-1,0)	0.62%	0.43%	0.80	12:10	0.80
(+1,+5)	0.50%	-1.15%	0.41	7:15	-1.33

$, (,) significant at .10 *, <, > significant at .05
, <<, >> significant at .01 *, <<<, >>> significant at .001

Figure 4.8
Eventus statements for event studies cumulating returns between paired event dates.

filename request 'G:\Some Folder\Filename.extension';
filename userstok 'G:\Some Folder\Filename.extension';

EVENTUS TWIN NONCRSP [**MONTHLY**|**WEEKLY**] [**PAGE=WIDE**];

[**TITLE** *'text'*;]

[**TITLE2** *'text'*;]

REQUEST [**ID=***variable* **IDFMT=***format*]
[**DATEFMT=MMDDYY**|**YYMMDD**|**DDMMYY**|**DATE**|**CRSP**]
[**EST=**−*value*|+*value*] [**POOL**] [**ESTLEN=***n*] [**NAME**];

WINDOWS [**EVENT1=***descriptor1* **EVENT2=***descriptor2*];

EVTSTUDY [**NOPLIST**] [**DETAIL**]
[**NOSTD**] [**VALUE**|**BOTH**] [**SW**|**GARCH**|**EGARCH**]
[**CSECTERR**] [**STDCSECT**] [**SERIAL**] [**MEDIAN**] [**BUYHOLD**]
[**TAIL=1**|**2**] [**OUTSAS=***libref.membername]*;

on the `EVENTUS` statement and the specification of the `WINDOWS` statement. The "descriptors" in the `WINDOWS` statement are 1–11 character names by which you want Eventus to designate the two event dates on the output. Blank spaces are not allowed within either descriptor. See Section 4.1 for descriptions of the remaining options.

Each line of the request file must contain a pair of event dates, separated by one or more blank spaces, in the location where the single event date goes in a normal event study request file.

Chapter 5

Extracting Event Study Results for Further Analysis

This chapter describes how to use the EXTRACT statement. The EXTRACT statement selects and organizes daily or monthly firm-by-firm cumulative or buy-and-hold compounded abnormal returns from results saved by the EVTSTUDY statement option OUTSAS. The most common use of EXTRACT is to create a data file for cross-sectional analysis of abnormal returns.

Be careful to distinguish EXTRACT from RETURNS. RETURNS, covered in Chapter 7, selects returns directly from the CRSP database and stores them in a disk file. It is intended mainly for those researchers who want to analyze returns entirely with their own programs. EXTRACT is useful for those who want to take advantage of the event study features of Eventus, then perform their own cross-sectional analyses.

Figure 5.1 displays the Eventus statements needed to extract and cumulate saved event study results. The statements and their options are explained below.

5.1 The EVENTUS Statement

The EVENTUS statement is required. It sets up internal parameters for your Eventus run. No options are needed on the EVENTUS statement.

Figure 5.1
Eventus statements for extracting saved event study results.

EVENTUS ;

WINDOWS *(begin,end)* [*(begin,end) ...*]**;**

EXTRACT INSAS=*libref.membername*
[**ID**=*variable* **IDFMT**=*format*] [**VPREFIX**=*prefix*]
[**WPREFIX**=*prefix*] [**TYPE**=[**CP**|**RAW**|**MAR**|**SW**]]
[**VALUE**] [**EXTEND**[=*n*]] [**SERIAL**] [**BUYHOLD**]
[**TEXT**|**BINARY**]
[**OUTSAS**=*libref.membername*|**EXTFILE**=*fileref*]**;**

5.2 The WINDOWS Statement

Use the WINDOWS statement to specify one or more intervals, or "windows", of days, weeks or months (the data frequency that the event study used.) The cumulative or compounded abnormal return for each window is calculated and stored for each firm. You can list up to 99 windows on one WINDOWS statement. The windows must be within the PRE and POST limits of the original event study. Within the limits, the windows need not be the same as those listed in the WINDOWS statement, if any, that preceded the EVTSTUDY statement. When the saved data are from a TWIN event study, omit the WINDOWS statement. Otherwise, your program must include exactly one WINDOWS statement before the EXTRACT statement.

5.3 The EXTRACT Statement

Identifying the saved event study file

Use INSAS to tell Eventus where to find the saved event study data. The *libref* and *membername* should match the OUTSAS specification of the original EVTSTUDY statement. (See page 31 in Chapter 3.)

Selecting a stored identification variable

If the EVTSTUDY program that saved the data included ID and IDFMT on the REQUEST statement, then you can repeat them on the EXTRACT statement. If you do, the identifying variable values will be added to the OUTSAS data set or the EXTFILE file created by EXTRACT.

Naming the window variables and selecting weights

The VPREFIX option selects a prefix for the variable names under which the window cumulative abnormal returns are stored in the OUTSAS data set. The prefix may be up to 6 characters long, of which the first must be a letter or underscore. Eventus completes the variable name by appending an integer from 1 to 20 indicating the position of the window on the WINDOWS statement. For example, if you specify VPREFIX=DEP and

```
WINDOWS (−1,0) (−2,2);,
```

the OUTSAS data set will include a variable named DEP1 containing the days (−1,0) cumulative abnormal return for each firm. The cumulative abnormal return for the second window, (−2,2), will be a variable named DEP2, and so on. The default is VPREFIX=WINAR.

The WPREFIX option specifies that you want the output to include a weight variable and gives the prefix for the weight variable name. The weight variable is the reciprocal of the variance of the market model cumulative abnormal return. Running a squares regression with the weight variable specified in the WEIGHT statement of PROC REG, is equivalent to estimating an ordinary least squares regression with all the variables (including the vector of ones used for the intercept) multiplied by the square root of the weight variable.[1] You can specify WPREFIX if you use TYPE=MM (see below) and the original EVTSTUDY statement did not specify NOMM or NOSTD, or with another TYPE value if the original EVTSTUDY statement specified STDALL.

When saving cumulative abnormal returns in an external file instead of a SAS data set, omit VPREFIX. However, it is still necessary to specify WPREFIX=W (the argument is arbitrary) to generate weights, even though the weight prefix will not appear in the external file. In the external file, the cumulative abnormal returns for the first firm appear on one line, continuing

[1] See Draper and Smith (1981), Section 2.11; or Neter, Wasserman and Kutner (1983), pp. 171–172.

onto additional lines as needed. Then the weights for the first firm appear on a new line, or more than one line if necessary. The returns for the second firm then start on a new line, and so on.

When extracting data from a `TWIN` event study, omit `VPREFIX` even if you are creating a SAS data set. `EXTRACT` assigns the name `CAAR` to the cumulative abnormal return from a `TWIN` event study.

Selecting the type of abnormal return to extract

By default, Eventus extracts market model returns unless the event study program used the CRSP excess return file. Alternatively, you may specify `CP`, `RAW`, `MAR` or `SW` to get comparison period, raw, market adjusted, or Scholes-Williams returns, provided that the original event study included the type of abnormal return you specify. Note that `EXTRACT` considers `RAW` to be a type of abnormal return. In order to extract raw returns, the `EVTSTUDY` statement in the event study program must include the `RAW` option and the package specifier `D`.

Extending a window to make up for missing days

If you specify the `EXTEND` option, Eventus will attempt to make up for missing returns within the window. For example, suppose you want to output a two-day window for a sample of takeover targets, some of which experienced trading halts on day -1 or 0, or both. You could specify `EXTEND=3` to attempt to obtain 2 days' worth of abnormal returns for each firm. In this case, if one of days -1 and 0 were missing, Eventus would extend the window to day $+1$. If day $+1$ were also missing, the window would be extended to day $+2$ for that firm. Eventus keeps trying to extend the window until it has obtained the "normal" number of returns for the window, or until it has exhausted the n days following the window. Specifying `EXTEND` without `=`n is equivalent to specifying `EXTEND=1`. You may want to consider the use of weighted least squares regression with abnormal returns generated using this option.

Adjusting the weights for serial correlation

If you ran the original event study with the `SERIAL` option described on page 29 (or the `STDCSECT` option, which implies `SERIAL`), Eventus adjusted the window test statistics to reflect the serial correlation that is inherent

in abnormal returns. To make such an adjustment to the weights for cross-sectional analysis, specify the SERIAL option on the EXTRACT statement. This option only is relevant when you use the WPREFIX option. The default is not to make any adjustment to the weights for serial correlation, even though the test statistics for the original event study incorporated the adjustment.

Computing buy-and-hold compounded returns

To obtain buy-and-hold abnormal returns for windows, specify BUYHOLD on the EXTRACT statement. This feature requires that the BUYHOLD option also appeared on the EVTSTUDY statement. The default is to generate cumulative abnormal returns even when the original event study used buy-and-hold returns.

5.4 Usage example

Figure 5.2 displays the Eventus statements to perform an event study using the request file from Chapter 2 with daily CRSP data. In this example, no WINDOWS statement happens to appear before the EVTSTUDY statement, so Eventus generates default windows as explained in Chapter 3. The EVTSTUDY statement includes two option specifications to save needed data in a SAS data set. The option OUTSAS=WORK.INTERMEDIATE names the SAS data set to be created.[2] Using WORK as the first part of a two part data set name specifies a temporary data set. A temporary data set ceases to exist after the user closes SAS in the case of interactive use, or after execution completes in the case of a batch run. Eventus currently requires the two part name, but WORK.INTERMEDIATE and the one part name INTERMEDIATE are completely interchangeable in ordinary SAS language and procedures. A permanent data set could have been specified by replacing WORK with a libref previously defined in a libname statement or the Add New Library dialog.

The option PACKAGE=DG specifies that the contents of the data set will include the necessary data to build firm-by-firm cumulative abnormal returns (D) and the corresponding weighted least squares (WLS) weights (G.) A complete list of PACKAGE option values appears in Table B.1 in Appendix B.

[2] If Eventus were running in SAS version 6.12 or earlier, the user would have to choose a shorter data set name; INTERMEDIATE would trigger an error because it is longer than eight characters.

Figure 5.2
Example using EXTRACT to organize firm-by-firm event study results for further analysis.

```
filename request 'F:\Any Folder\Filename.extension';
eventus;
 title 'US Targets of Canadian Acquirers 1997-1998';
 request;
 evtstudy outsas=work.intermediate package=DG;

 windows (-30,-2) (-1,0);
 extract type=MM vprefix=wincar wprefix=weight
   insas=work.intermediate outsas=work.abnormalreturns;
```

A new WINDOWS statement must come between the EVTSTUDY and EXTRACT statements, whether or not there is a WINDOWS statement before EVTSTUDY. The windows listed on the new WINDOWS statement can be different from those on any preceding WINDOWS statement.

The EXTRACT statement includes options to specify the type of abnormal return benchmark to use (market model), the prefix (wincar) to use in building variable names for the cumulative abnormal returns (CARs), the prefix for variable names for the WLS weights (weight), the name of a SAS data set previously built by an EVTSTUDY statement (work.intermediate), and the name of the output SAS data set to create (work.abnormalreturns.)

Figure 5.3 displays the contents of work.abnormalreturns produced by the SAS statements proc print data=abnormalreturns; id permno. The _weight_ variable has the value Equal for all observations in the example because only the event study use the default equal weighted market index. Had the BOTH option appeared on the EVTSTUDY statement, there would have been two observations in work.abnormalreturns for each PERMNO, one with _weight_ of Equal and one with Value. The CAR-WLS weight pair wincar1, weight1 corresponds to the first window listed on the last WINDOWS statement, (−30,−2) in the example, and the second pair corresponds to the second window. To conform to the requirements of the weight statement in the SAS regression procedure proc reg, the weights are reciprocals of variance, not portfolio weights.

Assume that the researcher creates a SAS data set explanatory, with one observation for each firm in the sample of various explanatory variables, identified by PERMNO. The researcher can then merge the two data sets and

Figure 5.3
Contents of SAS data set `abnormalreturns` produced by Figure 5.2 code.

US Targets of Canadian Acquirers 1997-1998 7

PERMNO	_weight_	wincar1	weight1	wincar2	weight2
10506	Equal	0.56738	5.777	-0.03257	83.88
10914	Equal	0.13304	55.647	0.41793	807.09
36150	Equal	0.20945	91.579	0.19949	1331.05
67652	Equal	0.29229	46.506	0.44130	675.01
72100	Equal	-0.08272	21.992	0.81411	319.29
75111	Equal	0.52603	37.952	0.10444	552.85
75241	Equal	-0.04506	88.500	-0.04199	1241.09
76263	Equal	0.03851	209.194	-0.02503	3045.64
76369	Equal	-0.01549	2.847	1.34032	41.35
76754	Equal	0.49761	30.693	0.14292	438.04
77142	Equal	0.10692	41.911	0.06721	600.09
77170	Equal	0.54004	6.870	-0.35172	100.62
77446	Equal	-0.08327	19.854	0.22735	290.88
77833	Equal	-0.07545	13.418	0.22582	191.68
79739	Equal	0.10539	44.350	0.39443	644.39
83447	Equal	0.29044	27.797	-0.08052	396.70

estimate a cross-sectional regression with statements like the following,

```
data regression_variables;
 merge abnormalreturns explanatory;
  by permno;
proc reg data=regression_variables;
 model wincar2=regressors;
 weight weight2;
```

where the word `regressors` is replaced by the name of one or more explanatory variables. The `weight` statement can be omitted for ordinary least squares regression. The procedure allows many additional options, including hypothesis tests using a heteroscedasticity consistent covariance matrix. Please see SAS documentation for further details.

Chapter 6

Event Studies Using the Event Parameter Approach

In the conventional approach, the market model or other benchmark parameters are estimated over a period that excludes the event dates to be tested. The abnormal returns on the event dates then are estimated in a second stage. In the event parameter approach, the market model is augmented by adding dummy variables to identify event periods, allowing the joint estimation of the market model parameters and abnormal returns. Karafiath (1988) provides a tutorial on the event parameter approach; Malatesta (1986) describes the approach using joint generalized least squares estimation.

In principle, there can be a unique dummy variable for each date in the event period, a dummy variable for each of several event dates, or a common dummy variable for a range of dates. Eventus implements the event parameter approach by defining a base event date (day or month 0) as in the conventional approach, then creating a dummy variable for each window listed on the `WINDOWS` statement. Since a window can consist of one or more dates relative to day 0, the researcher has considerable flexibility in defining the dummy variables.

6.1 Statements for the Event Parameter Approach

Figure 6.1 displays the Eventus statements to run a single-event date event study. The options that are specific to the event parameter approach are

Figure 6.1
Eventus statements for an event parameter approach event study.

EVENTUS [**NONCRSP**] [**MONTHLY**] [**PAGE=WIDE**];

[**TITLE** *'text'*;]

[**TITLE2** *'text'*;]

REQUEST [**CUSIP**|**CUSIPERM**] [**ID=***variable* **IDFMT=***format*]
[**DATEFMT=MMDDYY**|**YYMMDD**|**DDMMYY**|**DATE**]
[**EST=**−*value*| +*value*] [**POOL**] [**ESTLEN=***n*] ;

WINDOWS *(begin,end) [(begin,end) ...]*;

EVTSTUDY [**NONAMES**] **OLSPARAM**|**SUR**|**ITSUR**
[**FACTORS=***n*] [**VALUE**]
[**PRE=***periods*] [**POST=***periods*] ;

described below.

The EVENTUS statement

After any needed `filename` or `libname` statements, an Eventus program starts with an `EVENTUS` statement. The event parameter approach can be used either with direct CRSP database retrieval or with the `NONCRSP` option. For more details on direct CRSP access and the options that can be used with it, please see Chapter 3. For more information on event studies with non-CRSP data, please see Chapter 4.

The REQUEST statement

The `REQUEST` statement instructs Eventus to read the request file and, for `NONCRSP` event studies, the user-supplied returns file. The request file is an external file; you use it to supply the PERMNO or CUSIP identifiers for your sample, the event dates, and optional other information.

Each line of the request file contains data for a single security-base date combination. The base date is used as day (or month) 0 in the event study. The base date in the event parameter approach is not necessarily an event

date. It is simply a reference point around which the event windows will be built. At a minimum, each line must contain a PERMNO or CUSIP and base date. For example, a line of the request file could look like this:

```
10123  19900512
```

Spacing does not matter as long as the PERMNO or CUSIP comes first, then the base date, with at least one space between. If the request file contains CUSIPs, you need to specify either the `CUSIP` option or the `CUSIPERM` option. Use the `CUSIPERM` option when directly accessing a CRSP database that is sorted by PERMNO, to instruct Eventus to convert the CUSIPs in the request file to PERMNOs. Use the `CUSIP` option when Eventus is not reading directly from a CRSP database or when the CRSP data are sorted by CUSIP.

The rest of this section describes `REQUEST` statement options for the processing of dates and market and stock returns and for the construction of the estimation period. A SAS `filename` statement or the operating system must associate the fileref `REQUEST` with the request file.[1]

Including an identification variable

Each line of your request file may include an identification variable following the base date. Eventus will read it, and print it on subsequent output, if you name it with `ID=`. `ID=ID` works, or you can choose another name. Use `IDFMT` to tell Eventus what format to use for reading and printing the identifying variable. `IDFMT=4` means a 1–4 digit integer, while `IDFMT=$4` means a four-character string. Other lengths and other SAS formats are permitted also. With `IDFMT=$4` specified, a line of your request file might look like this:

```
16372210 19840920 AY4E
```

Using date formats other than YYMMDD

The `REQUEST` statement allows you to specify the format of dates in your request file. You may list calendar dates in nearly any conventional format. If you use either six- or eight-digit YYMMDD, the default, you need not specify `DATEFMT`. Besides MMDDYY and DDMMYY, you can use the SAS date format DATE. (An example of a date in DATE format is 19OCT97.)

[1] If you are familiar with permanent SAS data sets, you might want to read about the `INSAS` option on page 147.

Options for constructing the estimation period

The estimation period is simply a time period that contains no event dates. Eventus will combine the estimation period with the event period to estimate the augmented market model. The estimation and event periods may be defined so that they are discontinuous if desired.

EST= and POOL By default, Eventus determines the estimation period for each firm by subtracting 46 trading days from the base date in your request file. The resulting date becomes the *last* day of the estimation period. If you want the estimation period moved back, say to 90 days before the base date, specify EST=−90. For an estimation period following the base date, specify a positive number. EST=+61 (the plus sign is optional) produces an estimation period that *begins* on day +61.[2]

The default estimation period for weekly event studies ends with week −10; for monthly event studies, the default is to end the estimation period with month −13.

To split the estimation period between pre- and post-base dates, specify POOL on the REQUEST statement. Then Eventus will chop your estimation period into two equal halves. For example,

```
REQUEST ...  EST=50 POOL;
```

defines an estimation period of which the first half ends with day −50, and the second half begins on day +50. The POOL option has no particular meaning in a non-CRSP job, since the researcher must have assembled the estimation period data previously.

EST has no effect on the length of the estimation period. The ESTLEN option, described next, changes the estimation period length.

ESTLEN= By default, the estimation period is 255 days long when using daily data and 52 weeks or 60 months long when using weekly or monthly data. You can change the estimation period length with ESTLEN. The largest number you can use is 999 and the smallest is 3. Eventus assumes that you are specifying the number of months for a monthly event study, weeks for a weekly event study, or days for a daily event study.

[2] It also is possible to specify the estimation period by calendar date instead of relatively; see EST=SPECIFIC on page 146 in Appendix B.

`MINESTN=` Specify `MINESTN=`*n* to remove an observation from the sample if the stock has fewer than *n* days or months of return data in the estimation period. For example, `MINESTN=60` means that if the stock has fewer than 60 usable returns in the estimation period, that observation will be dropped. Eventus tells you when observations drop out of the sample. `MINESTN` must be less than or equal to `ESTLEN`.

The `WINDOWS` statement

The `WINDOWS` statement is required for the event parameter approach. Use it to specify ranges of dates — "windows" — relative to the base date, over which Eventus should compute abnormal return parameters and statistical significance tests. For example, when daily returns are used, the window (−2,+2) defines a five-trading day period, from two days before through two days after day 0. Up to 99 windows can be specified. Single-date windows may be specified by repeating the relative date. For example, (3,3) establishes a window containing only day +3. For each window, Eventus enters a dummy variable in the event parameter regression model. The dummy variable is equal to one on each date within the window and equal to zero on all other dates in the combined estimation and event period.

The `EVTSTUDY` statement

The `EVTSTUDY` statement tells Eventus to run the event study. The options permit you to change the default event period, vary the amount of printed output, and select a market index and estimation method.

Selecting the amount of printed output

`EVTSTUDY` produces a report of the results of reading your data. The report lists the PERMNO or CUSIP, identifying variable and event date from your request file, and the name of the issuer if direct CRSP access is used or you specified the `NAME` option on the `REQUEST` statement. The report also tells, for each firm, how many returns Eventus found in the estimation period and in the event period. If you prefer not to get this report, specify the `NONAMES` option.

Selecting the value weighted market index

With direct CRSP access, the default is to use the equal-weighted index with dividends from the CRSP index file. To use the value-weighted index, specify `VALUE` on the `EVTSTUDY` statement.

Specifying additional return-generation factors

To customize the return-generating model, specify the `FACTORS` option on the `EVTSTUDY` statement. For example, `FACTORS=2` specifies a two-factor model. The option is only available in `NONCRSP` mode and requires that the return data be supplied in the form of SAS data sets. The use of a customized return-generating model is illustrated in the example below.

Specifying the number of days or months in the event period

`EVTSTUDY` computes and reports abnormal returns for the event period, which is defined by default as days or weeks -30 through $+30$, or months -12 through $+12$. To change from these defaults, specify the number of days before and after on the `EVTSTUDY` statement, using `PRE` and `POST`. These options may be used singly or in combination. For example, `PRE=60` means that the abnormal returns are to start with day -60; unless `POST=` is also specified, the latter retains its default value.

If you specify `PRE=` or `POST=`, you may need to change the estimation period from the default. To change the estimation period, specify `EST=` on the `REQUEST` statement. Eventus will stop and print an error message if the event period defined by `PRE` and `POST` overlaps the default or specified estimation period.

Selecting an estimation method

To invoke the event parameter method, specify one of the `EVTSTUDY` statement options `OLSPARAM`, `SUR`, or `ITSUR`. The options indicate estimation using ordinary least squares, seemingly unrelated regressions (also called joint generalized least squares), or iterated seemingly unrelated regressions methods respectively.

Figure 6.2
Eventus program to assemble data for the event parameter example.

```
filename request 'c:\myproject\request.dat';
filename trates 'c:\myproject\rates.dat';
libname project 'c:\myproject\sasfiles';
data project.request;
 infile request;
 input cusip:$8. crspday1:4.0 eventdat:yymmdd. name:$10;
run;
eventus getdata;
 title 'Returns for Event Parameter Example';
 request insas=project.request cusiperm datefmt=crsp
  shift1=-105 ndays=111;
 returns index vsas outsas=work.stockret;
proc sort data=stockret;
 by date;
run;
data t_rates; /* read in separate file containing T-note yields */
 infile trates;
 input date:mmddyy8. treas1yr:5.2;
 factor2=dif1(treas1yr)/100; /* change in 1-yr T-note rate */
 drop treas1yr;
run;
data stockret;
 merge stockret(in=needed) t_rates;
 by date;
 if needed; /* keep only matched t-note dates */
 keep date permno cusip return market factor2;
run;
proc sort data=stockret out=project.stockret;
 by cusip date;
run;
```

Figure 6.3
Request file for the event parameter approach demonstration.

```
00620310 5432 19840216 Adams
04557310 5513 19840613 Associated
14348310 5538 19840719 Carnation
17444010 5630 19841128 Citizens
```

6.2 An Event Parameter Approach Example

The example includes a two-factor return-generating model. The `FACTORS` option is available only in `NONCRSP` mode, so we first retrieve the needed data from the CRSP database. Figure 6.2 shows the Eventus program to extract the returns, with additional SAS code to process the second return-generating factor. The `EVENTUS` statement needs only the `NONCRSP` option. The program sets up a permanent SAS data library with the libref `project`. The specific methods for establishing a permanent SAS data library vary by system.

The program in figure 6.2 includes a preliminary data step to read the request file into the permanent SAS data set `project.request`. The request file includes the event date in both CRSP format (`crspday1`) and YYMMDD format (`eventdat`.) This allows the use of trading days in the `REQUEST` statement, where the `SHIFT1` option specifies that the extracted returns should start 105 trading days before day 0 and extend a total of 111 days (implying that the return series ends on day +5.)

The request file contains CUSIPs, so the `REQUEST` statement includes the `CUSIPERM` option. Figure 6.3 displays the request file. The program merges the stock and market returns extracted by Eventus with the data set containing the second factor, creating the permanent data set `project.stockret`. The permanent data set will become input to the second Eventus program that actually runs the event study. The name of the first return-generating factor must be `market`, which also is the name automatically constructed by the `RETURNS` statement. The second factor name must be `factor2`; additional factor names follow the obvious pattern.

Figure 6.4 shows the Eventus program to complete the event study. Note that the `REQUEST` statement includes `EST` and `ESTLEN` parameters consistent with the length of return series extracted in the first program. The first 100 trading days of the 111 day return series for each stock are arbitrarily called the estimation period, but the model estimation uses the entire 111 days.

Figure 6.4
Eventus program to complete the event parameter example.

```
libname project 'c:\Research\Demo Project'; /* varies by system */
eventus noncrsp;
 title 'Event Parameter Approach Demonstration';
 request insas=project.request est=-105 estlen=100;
 windows (-5,-2) (-1,0) (1,5);
 evtstudy insas=project.stockret pre=5 post=5 olsparam factors=2;
```

The event study results appear in figures 6.5 and 6.6. Eventus prints the results on three pages, which are condensed into two here. For each window, Eventus reports an F-test for each of three null hypotheses: the conventional event study hypothesis that the cumulative average abnormal return is equal to zero, the hypothesis that all the firm firms in the sample jointly have cumulative abnormal returns equal to zero, and the hypothesis that all the firm firms in the sample have equal cumulative abnormal returns, but not necessarily equal to zero. When you run Eventus, the SAS log window or log file, depending on the mode of SAS operation, reports the completion of data steps and procedures that Eventus executes internally. Most Eventus users will find that these reports have little meaning for them and they may ignore most of what is in the log. However, it is still a good idea to look briefly through the SAS log for messages that begin with `EVENTUS NOTE`, `EVENTUS WARNING`, or `EVENTUS ERROR`. If you contact us for technical support to help resolve a problem with an Eventus run, please include the entire log window contents, or log file, resulting from a single run.

Figure 6.5
First part of the results for the event parameter approach demonstration.

```
                    Event Parameter Approach Demonstration                  2

                                          Cumulative      Median
                                           Average      Cumulative
                                           Abnormal      Abnormal
                          Window            Return        Return

                          (-5,-2)          -1.07%        -1.68%
                          (-1,0)            4.30%         3.28%
                          (+1,+5)           3.05%         4.21%

Test: (-5,-2) CAAR=0
      Numerator:   0.366193   DF:     1  F Value:   0.3662
      Denominator:        1   DF:   388   Prob>F:   0.5454

Test: (-5,-2) CAR=0 for all firm-events
      Numerator:   0.460447   DF:     4  F Value:   0.4604
      Denominator:        1   DF:   388   Prob>F:   0.7648

Test: (-5,-2) CAR is equal across firm-events
      Numerator:   0.608711   DF:     3  F Value:   0.6087
      Denominator:        1   DF:   388   Prob>F:   0.6097

Test: (-1,0) CAAR=0
      Numerator:   12.08822   DF:     1  F Value:  12.0882
      Denominator:        1   DF:   388   Prob>F:   0.0006

Test: (-1,0) CAR=0 for all firm-events
      Numerator:   4.846007   DF:     4  F Value:   4.8460
      Denominator:        1   DF:   388   Prob>F:   0.0008

Test: (-1,0) CAR is equal across firm-events
      Numerator:   3.614898   DF:     3  F Value:   3.6149
      Denominator:        1   DF:   388   Prob>F:   0.0134

Test: (+1,+5) CAAR=0
      Numerator:   2.326272   DF:     1  F Value:   2.3263
      Denominator:        1   DF:   388   Prob>F:   0.1280

Test: (+1,+5) CAR=0 for all firm-events
      Numerator:    3.19089   DF:     4  F Value:   3.1909
      Denominator:        1   DF:   388   Prob>F:   0.0135

Test: (+1,+5) CAR is equal across firm-events
      Numerator:   3.065976   DF:     3  F Value:   3.0660
      Denominator:        1   DF:   388   Prob>F:   0.0280
```

Figure 6.6
Remaining results for the event parameter approach demonstration.

Event Parameter Approach Demonstration 4

Day	Mean Abnormal Return	Median Abnormal Return
-5	1.01%	1.32%
-4	0.29%	-0.21%
-3	-1.34%	-1.17%
-2	-1.04%	-0.83%
-1	0.21%	0.43%
0	4.10%	3.52%
1	1.34%	-0.83%
2	-0.92%	0.23%
3	-0.26%	0.37%
4	2.46%	2.57%
5	0.43%	0.49%

Chapter 7

Obtaining Returns, Prices, Volume, Number of Trades and Shares Outstanding from the CRSP Database

Eventus provides statements to read selected stock and index returns, stock prices or trading volume data from the CRSP database for future analysis by non-Eventus software. The selected returns can go in to any type of file that SAS can handle, including SAS data sets and external files (for example, ASCII character files.) Explanations of the statements and their options appear below.

7.1 The EVENTUS statement

The option `GETDATA` must appear on the `EVENTUS` statement preceding any `RETURNS` , `PRICES` or `VOLUME` statement.

The default data frequency is daily. To use an installed monthly CRSP database, specify `MONTHLY` on the `EVENTUS` statement. For example,

```
EVENTUS MONTHLY;.
```

To specify landscape orientation of the printed output, use the `PAGE=WIDE` option on the `EVENTUS` statement; the default is portrait orientation.

Figure 7.1
Eventus statements to read stock returns from a CRSP database.

filename request 'G:\Some Folder\Filename.extension';

EVENTUS GETDATA [**PAGE=WIDE**] [**MONTHLY**|**EXCESS**];

REQUEST [**ID=***variable* **IDFMT=***format*]
[**DATEFMT=MMDDYY**|**YYMMDD**|**DDMMYY**|**DATE**|**CRSP**]
[**AUTODATE**] [**NDAYS=***n*] [**NODIVIDX**] [**SP500**|**COMPOSIT**];

RETURNS [**BINARY**|**VSAS**|**HSAS**] [**INDEX**]
[**VALUE**|**BOTH**] [**SIC**] [**SHRCODE**]
[**EXTFILE=***fileref*|**OUTSAS=***libref.SAS_dataset*];

You may also select excess returns (based upon beta decile portfolios or standard deviation decile portfolios) if the appropriate CRSP add-on module is installed. To do so, specify `EXCESS` on the `EVENTUS` statement.

7.2 The `REQUEST` statement

The `REQUEST` statement instructs Eventus to read your request file, which is an external file containing PERMNO identifiers, dates, and sometimes other information. This section describes several `REQUEST` statement options for processing dates and returns. If you are familiar with permanent SAS data sets, you might want to take advantage of the `INSAS` option; see page 147. Otherwise, a SAS `filename` statement or the operating system must associate the fileref `REQUEST` with the request file.

Each line of the request file for a `RETURNS`, `PRICES` or `VOLUME` program should contain the following items: PERMNO, starting date of data to retrieve, ending date of data to retrieve, and identifying variable value (if any.) Omit the ending date if you specify `NDAYS=`, explained below, on the `REQUEST` statement. When reading a monthly database, the dates in the request file may be any day of the month.

Figure 7.2
Eventus statements to read stock prices from the CRSP database.

```
filename request 'G:\Some Folder\Filename.extension';
EVENTUS GETDATA [PAGE=WIDE] [MONTHLY];
    REQUEST [ID=variable IDFMT=format]
        [DATEFMT=MMDDYY|YYMMDD|DDMMYY|DATE|CRSP]
        [AUTODATE] [NDAYS=n];
    PRICES [BINARY|VSAS|HSAS] [SIC] [SHRCODE]
        [SHARES] [DISTRIB] [BIDASK [NOCLOSE]]
        [NMS [TRADES]] [SPLITADJ]
        [EXTFILE=fileref|OUTSAS=libref.SAS_dataset];
```

Including an identification variable

Each line of your request file may include an identification variable following the event date. Eventus will read it, and store it with the returns, prices or volume, if you name it with `ID=`. `ID=ID` works, or you can choose another name. Use `IDFMT` to tell Eventus the format to use for reading and printing the identifying variable. `IDFMT=4` means a 1–4 digit integer, while `IDFMT=$4` means a four letter word. Other lengths and other SAS formats also are valid.

Options for processing dates

The `REQUEST` statement allows you to specify how Eventus should handle the dates in your request file.

`DATEFMT=`

Calendar dates You can list calendar dates in nearly any conventional format. If you use either six- or eight-digit YYMMDD, the default, you do not

Figure 7.3
Eventus statements to read trading volume data from the CRSP database.

filename request 'G:\Some Folder\Filename.extension';

EVENTUS GETDATA [**PAGE=WIDE**] [**MONTHLY**];

REQUEST [**ID=***variable* **IDFMT=***format*]
[**DATEFMT=MMDDYY|YYMMDD|DDMMYY|DATE|CRSP**]
[**AUTODATE**] [**NDAYS=***n*];

VOLUME [**BINARY|VSAS|HSAS**] [**SIC**] [**SHRCODE**]
[**SHARES**] [**TRADES**] [**SPLITADJ**]
[**EXTFILE=***fileref*|**OUTSAS=***libref.SAS_dataset*];

need to specify `DATEFMT`. Besides MMDDYY and DDMMYY, you can use the SAS date format DATE.[1]

CRSP Trading Day Numbers Eventus never requires you to manually convert calendar dates to CRSP day numbers. If you happen to have CRSP day numbers, you can use them in your request file. Specify `DATEFMT=CRSP` on the `REQUEST` statement.

`AUTODATE` If some of the calendar dates in your request file may not be trading days, you can specify `AUTODATE` on the `REQUEST` statement. `AUTODATE` tells Eventus to convert automatically all calendar dates to trading days. Non-trading days are converted to the following trading day. For example, Eventus changes a Sunday to the following Monday, or Tuesday if Monday is a holiday. The `AUTODATE` option is ignored when reading a monthly file and when you specify `DATEFMT=CRSP`.

`NDAYS=` Specify `NDAYS=`*n* when you want the same, fixed number of trading days, weeks or months of returns, prices or volume data for every firm in the request file. Omit the ending date from your request file if you specify this option. The valid range of values for *n* is from 1 to 9999, but some computers will be unable to handle values above 1000 or so. The main constraint on

[1] An example of a date in DATE format is 19OCT1987.

the number of trading days, weeks or months of returns is the amount of space in the default WORK SAS data library. See the *SAS Companion* for your environment or ask a consultant at your site if you need to enlarge the WORK library.

Two additional date processing options, SHIFT1= and SHIFT2=, are available. These options shift the dates from the request file by a specified number of days. See page 150 in Appendix B.

Market index options

NODIVIDX Eventus normally supplies the returns including dividends on the equally weighted and value weighted (if the VALUE option is specified) indexes from the CRSP index file. Specify NODIVIDX to instruct Eventus to supply the index returns excluding dividends.

SP500 **and** COMPOSIT The CRSP NYSE-AMEX and NYSE-AMEX-Nasdaq index files contain the Standard and Poor's 500 Composite Index in addition to the equally weighted and value weighted indexes. Specify SP500 to tell Eventus to read the Standard and Poor's index return instead of the value weighted index. The CRSP Nasdaq index file includes the Nasdaq Composite Index return instead of the Standard and Poor's index. You instruct Eventus to read the Nasdaq Composite Index instead of the value weighted index when you specify COMPOSIT. Eventus does not determine whether the index file is actually the NYSE-AMEX(-Nasdaq) or the Nasdaq-only file.

7.3 The RETURNS, PRICES and VOLUME statements

Use only one of the RETURNS, PRICES or VOLUME statements in a single program. The following options are available.

Selecting an output file format

Five output formats are available: text, horizontal text, binary, vertical SAS data set, and horizontal SAS data set. The default text format can be printed, viewed on a display screen, and processed by SAS and other programs. The binary format, specified by the word BINARY on the RETURNS, PRICES or VOLUME statement, can be processed by programs. The vertical and horizontal

SAS data sets can be read only by SAS procedures and data steps. In a vertical data set, data are stored under a single variable name, `RETURN`, `PRICE`, or `VOLUM`, with one SAS observation per trading day or month per firm. In a horizontal format data set, there is a separate variable for each trading day or month, so that there is one SAS observation per firm. (See Appendix B, pages 141 and following for more detailed descriptions of the formats.)

Obtaining standard deviation portfolio excess returns

The `SPORT` option applies only to the `RETURNS` statement, not the `PRICES` and `VOLUME` statements, and only when the `EXCESS` option appears on the `EVENTUS` statement and the appropriate CRSP add-on module is installed. There are two kinds of returns. The first, which Eventus selects by default, is based on portfolios of stocks ranked by beta. The other is based on a ranking by standard deviation of return. To use standard deviation excess returns instead of beta excess returns, specify `SPORT` on the `RETURNS` statement.

Reporting market index returns with stock returns

These options are valid only on the `RETURNS` statement. Normally, no market index returns are included. The `INDEX` option specifies that you want returns on a market index for the same days as stock returns. By default, the index is the CRSP equally weighted index. To get value weighted or both equally and value weighted index returns, also specify `VALUE` or `BOTH`. The Standard and Poor's index return or the Nasdaq Composite Index return replaces the value weighted index when you specify `SP500` or `COMPOSIT` on the `REQUEST` statement.

Obtaining SIC codes

The `SIC` option causes Eventus to report SIC (industry classification) codes from the CRSP database. If an output SAS data set is being created (see the `OUTSAS` option below), the numeric variable `SICCODE` is included in the data set. The SIC code for the last reporting date on or before the starting date of data to retrieve is listed. An exception occurs when the earliest reporting date follows the starting date of data to retrieve; then the earliest SIC code is listed. See the latest CRSP manual for information about the accuracy of the SIC code data.

Obtaining share type codes

To store the CRSP two-digit share type code in an output SAS data set being created, specify the `SHRCODE` option. Normally the share code output will be the last share code reported on the CRSP database on or before the starting date of data to retrieve. In rare cases, the earliest share code reporting date for a stock may be later than the starting date of data to retrieve. In this case, Eventus uses the earliest share code. Please see the latest CRSP *Data Description Guide* for further details on the share type code.

Selecting bid and ask or intraday high and low prices

The `BIDASK` option, valid only on the `PRICES` statement, reads the secondary price variables — *Bid or Low Price* and *Ask or High Price* — from the CRSP stock database. CRSP documentation indicates that these prices may be either closing bid and ask prices, or intraday high and low transaction prices. Eventus does not distinguish between the two types of data when executing a `PRICES BIDASK` statement. However, the nature of the secondary prices can be inferred from the sign of the primary price variable when using daily data. CRSP uses a negative price to indicate a bid-ask average; in this case, the secondary price variables are closing bid and ask price quotes. When the primary price is positive, it is a closing trade price, and the secondary price variables are the lowest and highest trade prices during the day.

The secondary price data are stored in variables named `BIDLO` and `ASKHI` if you specify `VSAS`, or `BIDL1`–`BIDL`*nnnn* and `ASKH1`–`ASKH`*nnnn* if you specify `HSAS`. If you specify `BINARY` or use the default text output format, each firm's secondary prices follow its primary prices on separate lines identified by the words `BIDLO` and `ASKHI`.

The `NMS` option on the `PRICES` statement causes Eventus to read the Supplemental Nasdaq Data Arrays of the CRSP database, which report closing bid and ask quotes and number of trades.[2] When you specify both the `BIDASK` and `NMS` options, Eventus attempts to supply as many true bid and

[2]To use the `NMS` option with SFA format CRSP data files, a SAS `filename` statement, or the operating system, must associate the fileref `CRSPNMS` with the Supplemental Nasdaq File. If your system permanently stores the CRSP stock databases in SAS data sets, do not use the `filename` statement. Instead, use a `libname` statement to point to the location (SAS data library) of the Supplemental Nasdaq File, and give the libref using the `NLIBNAME` option on the `EVENTUS` statement. If it is the same libref as in the `LIBNAME` specification, you still must repeat it in the `NLIBNAME` option.

ask quotations instead of intraday high and low transaction prices as possible. Eventus reports bid and ask prices from the Supplemental Nasdaq Data Arrays when they exist, and secondary price data from the main time-series arrays otherwise. Nasdaq stocks not represented in the Supplemental Nasdaq Data Arrays generally have closing bid and ask quotations in the main time-series arrays. If the sample contains both Nasdaq and exchange-listed stocks, the `BIDASK NMS` option combination is likely to result in a mixture of bid-ask quotations and intraday high-low transaction prices.

To prevent Eventus from reading the closing transaction (primary) prices in addition to the secondary prices, specify `NOCLOSE` on the `PRICES` statement. Thus, the option combination `BIDASK NOCLOSE` reports only secondary prices from the main time-series arrays. An additional feature of the `NOCLOSE` option is that, when combined with the `NMS` option, it disables the mixing of Supplemental Nasdaq bid and ask quotations with secondary prices. Instead, Eventus reports both bid and ask prices from the Supplemental Nasdaq Data Arrays and secondary prices from the main time-series structures, *as four separate variables,* when you specify `BIDASK NMS NOCLOSE`.

Reporting the number of shares outstanding with stock prices or volume

The `SHARES` option is valid only on the `PRICES` and `VOLUME` statements. The `SHARES` option tells Eventus to store the number of shares outstanding, in thousands, along with the share price or trading volume.

In a text format file (the default output format) the number of shares appears after the price or volume, on the same line. In a binary format output file, the number of shares appears immediately following the price or volume in the final (next to final if `TRADES` is specified) four bytes or character positions of the output record in `IB4.` format (like a FORTRAN `INTEGER*4` variable.) In a `VSAS` or `HSAS` output file the number of shares is a variable named `SHARES`.

Deriving cash distributions

The `DISTRIB` option on the `PRICES` statement tells Eventus to read dividends and other cash distributions from the CRSP distribution structure. Daily or monthly per-share cash distributions are reported, with zero reported on any

non-ex-dividend date. If the SPLITADJ option is in effect, distributions are split-adjusted in the same manner as the price. In a VSAS or HSAS output file the cash distribution amount is a variable named DIVAMT, and the CRSP distribution code is a variable named DISTCODE.

Reporting the number of trades for Nasdaq stocks

The TRADES option is valid on the PRICES and VOLUME statements. Eventus reads the number of trades from the Supplemental Nasdaq File, which must be associated with the fileref CRSPNMS. When you use the PRICES statement you must also specify the NMS option for TRADES to work.

In a VSAS output file the number of trades is a variable named TRADES; in an HSAS file the variable names are TRAD1–TRAD*nnnn*.

Reporting Nasdaq-specific data items

The NASDINFO option is valid only on the VOLUME statement and only when the SHARES option is valid; NASDINFO automatically specifies SHARES also. The option reads the trading status trait code, the National Market System indicator and the number of market makers from the Nasdaq information structure array of the CRSP Nasdaq stock file. Eventus stores the data if the HSAS or VSAS option appears on the VOLUME statement. The variable names in the SAS data set are TRAIT, NMS and MAKERS. Eventus does not place the NASDINFO data in an external file if the default external text file output or BINARY output is selected.

Adjusting for splits and other stock distributions

By default, Eventus does not adjust stock prices, bid and ask quotations, trading volume, or shares outstanding for stock splits and stock dividends. (Returns on the CRSP files already are adjusted.) You can specify SPLITADJ on the PRICES or VOLUME statement to adjust for any stock splits or stock dividends that occur between the first and last dates of data. For example, suppose that a 2-for-1 stock split occurs on the 50th of 100 days of data. Without the SPLITADJ option, Eventus will report the actual data for each of the 100 days. With the SPLITADJ option, the trading volume and shares outstanding for the second 50 days will be divided by 2, and the prices and bid-ask quotations for the second 50 days will be multiplied by 2.

Selecting an output file location

Text or ASCII binary output is directed by default to the fileref `USERDATA`. If the fileref is undefined, a file named `userdata.dat` is usually created in the current working directory. To select a specific disk file in which to store the data, use a `filename userdata` statement, before the `EVENTUS` statement, to give the path and name of the file to be created. To use a different fileref besides `userdata,` specify the fileref with the `EXTFILE` option.

Vertical and horizontal SAS data sets require a libref to point to the aggregate storage location, typically a folder or directory, where the data set should go. If you specify the `VSAS` or `HSAS` option, also specify the two part SAS data set name (*libref.membername*), using the `OUTSAS` option. For a temporary data set (which will no longer exist after the user closes SAS or after the batch run completes), use the libref `work`. For a permanent data set, use a `libname libref` statement, before the `EVENTUS` statement, to point to the aggregate storage location, replacing "libref" with the desired libref.

Chapter 8

Converting Calendar Dates to CRSP Trading Day or Month Numbers Using DATECONV

DATECONV is an Eventus statement that converts calendar dates to CRSP style trading day, week or month numbers. **It is not necessary to use DATECONV before running most Eventus applications.** Eventus automatically converts calendar dates to CRSP trading day numbers without explicit instructions. Only researchers who have special requirements for CRSP day numbers, perhaps for use with non-Eventus software, need DATECONV.

Figure 8.1 lists the Eventus statements needed to use DATECONV. The options are described below.

8.1 The EVENTUS statement

If you are converting pairs of dates for each PERMNO, specify TWIN. If you are converting dates for use with a monthly or weekly database, specify MONTHLY or WEEKLY on the EVENTUS statement. Eventus uses the CRSP database's calendar to determine the association between calendar dates and CRSP day, week or month numbers.

Figure 8.1
Eventus statements for converting calendar dates to CRSP trading day or month numbers.

filename request 'G:\Some Folder\Filename.extension';
filename userdata 'G:\Some Folder\Filename.extension';

EVENTUS [**TWIN**] [**MONTHLY**];

DATECONV [**CUSIPERM**] [**ID**=*variable* **IDFMT**=*format*]
[**INSAS**=*libref.membername*]
[**INSAS2**=*libref.membername*]
[**DATEFMT=MMDDYY|YYMMDD|DDMMYY|DATE|CRSP**]
[**AUTODATE**] [**NDAYS**=n] [**SHIFT1**=n_1] [**SHIFT2**=n_2]
[**OUTDTFMT=MMDDYY|YYMMDD|DDMMYY|DATE**]
[**OUTSAS**=*libref.membername*|**EXTFILE**=*fileref*];

8.2 The DATECONV statement

The DATECONV statement instructs Eventus to read your request file, which is an external file that contains PERMNO identifiers, dates, and sometimes other information.[1] A SAS filename statement or the operating system should associate the fileref REQUEST with your request file.

Each line of the request file for a DATECONV program should contain the following items: PERMNO, first date, second date if the option TWIN appears on the EVENTUS statement, and the identifying variable value (if any.) Omit the second date if you specify NDAYS=, explained below, on the DATECONV statement. When converting calendar dates to month numbers, the dates in the request file may be any day of the month. When converting to week numbers, each calendar date can be the last trading day in the week, or with the AUTODATE option (see below), any prior date in the week.

[1] If you are familiar with permanent SAS data sets, you may want to read about the INSAS option on page 147.

Searching by CUSIP or converting CUSIPs to PERMNOs

1993 and earlier editions of the CRSP stock files are sorted by CUSIP identifiers rather than permanent identification numbers. If you are using one of these files, provide CUSIPs instead of PERMNOs in the request files. You also need to specify the `CUSIP` option on the `REQUEST` statement. Follow the same request file format as described elsewhere in this chapter but supply an eight-character CUSIP instead of a numeric PERMNO for each stock.

Sites where the CRSP data exist as SAS data sets often index the data sets so that programs can search them by either CUSIP or PERMNO without sorting. The default in Eventus is to search by PERMNO. To search by CUSIP, provide CUSIPs instead of PERMNOs in the request files and specify the `CUSIP` option on the `REQUEST` statement.

If you have CUSIPs in the request file, you can specify the `CUSIPERM` option to convert them to CRSP PERMNOs. Eventus uses the PERMNOs to search the CRSP database and label the results, but does not change your request file. Only the PERMNOs, not the CUSIPs, appear in the output file.

The `CUSIPERM` option works properly only if the PermnoUp program described in the Eventus installation instructions is run once after each annual or quarterly update of the CRSP stock database.

Including an identification variable

Each line of your request file may include an identification variable following the event date. Eventus will read it, and store it with the PERMNO identifiers and converted dates, if you name it with `ID=`. `ID=ID` works, or you can choose another name. Use `IDFMT` to tell Eventus the format to use for reading and printing the identifying variable. `IDFMT=4` means a 1–4 digit integer, while `IDFMT=$4` means a four letter word. Other lengths and other SAS formats also are valid.

Using grouping variables, group weights and short-long indicators

If the input request file contains a grouping variable (see page 17), specify the `GROUP` option on the `DATECONV` to have Eventus include the grouping variable in the updated file. Add the `GRWEIGHT` option if the request file also contains a within-group weight for each observation. If the input request file

contains an S or an L to indicate short or long (see page 19), specify SHORT on the DATECONV statement. The S or L must follow the PERMNO, date(s), and any identifying variable, grouping variable, and group weight. These options have no effect on the operation of DATECONV except to copy the grouping variable, weights or short-long indicator to the output file. DATECONV statement processing makes no use of the grouping variable, group weight, or short-long indicator except to copy them from the input request file to the correct position in the updated request file.

Options for processing dates

The DATECONV statement allows you to specify how Eventus should handle the dates in your request file.

DATEFMT=

Calendar dates You can list calendar dates in nearly any conventional format. If you use either six- or eight-digit YYMMDD, the default, you don't need to specify DATEFMT. Besides MMDDYY and DDMMYY, you can use the SAS date format DATE.[2]

CRSP Trading Day Numbers To convert CRSP day numbers back to calendar dates, specify DATEFMT=CRSP on the DATECONV statement.

AUTODATE

If some of the calendar dates in your request file may be non-trading days, specify AUTODATE on the DATECONV statement. AUTODATE tells Eventus to convert automatically *all* calendar dates to trading days. Non-trading days are converted to the following trading day. For example, a Saturday would be changed to the following Monday, or Tuesday if Monday were a holiday. To convert non-trading dates to the previous trading date instead of the next, specify AUTODATE=BACK. The AUTODATE option has no effect when converting between calendar dates and CRSP month numbers, because all months within the range of the CRSP calendar are trading months.

[2] An example of a date in DATE format is 19OCT1987.

NDAYS=

Specify NDAYS=*n* if you want create a second date, to be the same, fixed number of trading days from the first date for every firm in your request file. Specify TWIN on the EVENTUS statement, but omit the second date from your request file, if you use NDAYS.

SHIFT1= and SHIFT2=

SHIFT1= and SHIFT2= shift the dates from the request file by a specified number of days. When the request file contains calendar dates, these options determine the number of calendar days by which to shift. When the request file contains CRSP trading day numbers, these options determine the number of trading days by which to shift. A positive shift adds days; a negative shift subtracts days.

OUTDTFMT=

Many DATECONV programs do not need this option. Use it only when you are converting from CRSP-style day, week or month numbers back to calendar dates. (In this case, DATEFMT=CRSP also should appear on the DATECONV statement.) This option specifies the calendar date format into which to convert CRSP trading day numbers. You do not need this option when your request file contains calendar dates, because in that case Eventus always converts to CRSP day numbers. You can specify any SAS date format. The default is YYMMDD.

Sorting the updated request file

The observations in the updated request file will appear in PERMNO order by default. To have the observations sorted by the identifying variable, include the SORTBYID option on the DATECONV statement. For this to work, the input request file must contain an identifying variable value on each line and you also must specify the ID= option.

Selecting an output file location

The converted request file is directed by default to the fileref USERDATA. If the fileref is undefined, a file named userdata.dat is usually created in the

current working directory. To select a specific disk file for the converted request file, use a `filename userdata` statement, before the `EVENTUS` statement, to give the path and name of the file to be created. To use a different fileref besides `userdata,` specify the fileref with the `EXTFILE` option. For maximum safety, do not specify the fileref `request` or the input request file may be overwritten.

You may also have Eventus store the converted dates in a SAS data set. Specify the two part SAS data set name *libref.membername*, using the `OUTSAS` option.

Chapter 9

Converting CUSIP Identifiers Using CUSIPERM

CUSIPERM converts 8-character common stock CUSIP values to the corresponding CRSP permanent identification number, or PERMNO. Before 1995 when the 1994 stock files appeared, CRSP sorted its files by the CUSIP at the ending date of the file or, if the stock went off earlier (for example, due to the company being taken over), the last CUSIP it had. This resulted in CUSIPs that could change from one annual edition of the files to the next. The CRSP stock databases currently are sorted in PERMNO order. The purpose of the CUSIPERM statement is to convert CUSIPs from any previous edition of the CRSP stock files to PERMNOs.

Whoever takes care of Eventus at your site should run the PermnoUp program, supplied with Eventus, to generate databases for CUSIPERM after installing Eventus and after each annual or quarterly update of the CRSP stock database.

This chapter describes the stand-alone version of CUSIPERM, which requires its own Eventus run. Use this version to create a copy of your request file that contains PERMNOs. There also is a CUSIPERM option on the REQUEST statement that converts CUSIPs "on the fly" during an event study or data extraction job. However, the REQUEST statement option version does not produce a converted version of the request file for use in future Eventus jobs.

Figure 9.1 lists the Eventus statements required to convert CUSIP identifiers. The statements are discussed below.

Figure 9.1
Eventus statements to convert CUSIP identifiers to CRSP permanent identification numbers.

EVENTUS ;

CUSIPERM [**COLUMN**=*n*] [**EXTFILE**=*fileref*];

9.1 The EVENTUS statement

The EVENTUS statement is mandatory. No options are needed on the EVENTUS statement for a CUSIPERM run.

9.2 The CUSIPERM statement

The CUSIPERM statement instructs Eventus to read your request file, which contains CUSIP identifiers. A SAS filename statement or an operating system control language statement must associate the fileref REQUEST with the request file. In a CUSIPERM program, the request file need not conform to the request file format described elsewhere in this guide. An eight character CUSIP identifier must appear within the first 80 columns of each line. The CUSIP may begin in any column position on the line. If the CUSIP is the first non-blank item on each line, it need not even start in the same column on each line. For example, it could be preceded by two blanks on some lines and no blank on others. Any other information you want may appear on the same line. CUSIPERM duplicates your original file exactly, except that it replaces any CUSIP with the corresponding five-digit PERMNO followed by three blanks and sorts the lines in PERMNO order. However, any information beyond column 80 will not be copied.

Make sure that no leading zeros are omitted from the CUSIPs. CUSIPs can contain letters as well as digits, so CUSIPERM treats the CUSIP as an eight-character string, not a numeric variable.

The following two options may be specified on the CUSIPERM statement.

In which column does the CUSIP identifier begin? The `COLUMN` option

The `COLUMN` option is needed only when non-blank characters precede the CUSIP identifier in the request file. Specify the column number of the first character of the CUSIP.

Selecting an output file location

The original request file remains unchanged. The converted request file containing PERMNOs is directed by default to the fileref `USERDATA`. If the fileref is undefined, a file named `userdata.dat` is usually created in the current working directory. To select a specific disk file in which to store the converted request file, use a `filename userdata` statement, before the `EVENTUS` statement, to give the path and name of the file to be created. To use a different fileref besides `userdata`, specify the fileref with the `EXTFILE` option. To avoid overwriting the input request file, do not specify the fileref `request`.

Appendix A

Technical Reference

A.1 Event Study Prediction Errors

Market Model Abnormal Returns

Assume that security returns follow a single factor market model,

$$R_{jt} = \alpha_j + \beta_j R_{mt} + \epsilon_{jt},$$

where R_{jt} is the rate of return of the common stock of the j^{th} firm on day t; R_{mt} is the rate of return of a market index on day t; ϵ_{jt} is a random variable that, by construction, must have an expected value of zero, and is assumed to be uncorrelated with R_{mt}, uncorrelated with $R_{kt,k\neq j}$, not autocorrelated, and homoscedastic. β_j is a parameter that measures the sensitivity of R_{jt} to the market index. Define the *abnormal return* (or *prediction error*) for the common stock of the j^{th} firm on day t as:

$$A_{jt} = R_{jt} - (\hat{\alpha}_j + \hat{\beta}_j R_{mt}),$$

where the coefficients $\hat{\alpha}_j$ and $\hat{\beta}_j$ are ordinary least squares estimates of α_j and β_j.

The *average abnormal return* (or *average prediction error*) AAR_t is the sample mean:

$$AAR_t = \frac{\sum_{j=1}^{N} A_{jt}}{N},$$

where t is defined in trading days relative to the event date (*e.g.* $t = -60$ means 60 trading days before the event.)

Over an interval of two or more trading days beginning with day T_1, and ending with T_2, the *cumulative average abnormal return* is

$$CAAR_{T_1,T_2} = \frac{1}{N}\sum_{j=1}^{N}\sum_{t=T_1}^{T_2} A_{jt}.$$

Over an interval of two or more trading days beginning with day T_1, and ending with T_2, the *average compounded abnormal return* is

$$ACAR_{T_1,T_2} = \frac{1}{N}\sum_{j=1}^{N}\left[\prod_{t=T_1}^{T_2}(1+R_{jt})-1\right] - \left[(1+\hat{\alpha}_j)^{(T_2-T_1+1)}-1\right] - \hat{\beta}_j\left[\prod_{t=T_1}^{T_2}(1+R_{mt})-1\right].$$

Scholes-Williams Beta Estimation

When the `SW` option appears on the `EVTSTUDY` statement, Eventus reports market model results using betas estimated by both ordinary least squares and the method of Scholes and Williams (1977). The Scholes-Williams beta estimator is

$$\hat{\beta}_j^{\star} = \frac{\hat{\beta}_j^{-} + \hat{\beta}_j + \hat{\beta}_j^{+}}{1+2\hat{\rho}_m},$$

where $\hat{\beta}_j^{-}$ is the OLS slope estimate from the simple linear regression of R_{jt} on R_{mt-1}, $\hat{\beta}_j^{+}$ is the OLS estimate from the regression of R_{jt} on R_{mt+1}, and $\hat{\rho}_m$ is the estimated first-order autocorrelation of R_m. As in OLS, the intercept estimator forces the estimated regression line through the sample mean:

$$\hat{\alpha}_j^{\star} = \overline{R_j} - \hat{\beta}_j^{\star}\overline{R_m}.$$

$\overline{R_j}$ is the mean return of stock j over the estimation period and $\overline{R_m}$ is the mean market return over the estimation period.[1]

[1] Eventus uses the simplifying assumption that the use of Scholes-Williams estimates does not affect the formula for $s^2_{A_{jt}}$ below. Analytically this assumption is not strictly correct, but simulation results obtained by the author show that tests using the assumption are well specified.

Market Adjusted Returns

Market adjusted returns are computed by subtracting the observed return on the market index for day t, R_{mt}, from the rate of return of the common stock of the j^{th} firm on day t:

$$A_{jt} = R_{jt} - R_{mt}.$$

The definitions of the average abnormal return and cumulative average abnormal return follow those for market model abnormal returns above.

Comparison Period Mean Adjusted Returns

Comparison period mean adjusted returns are computed by subtracting the arithmetic mean return of the common stock of the j^{th} firm computed over the estimation period, $\overline{R_j}$, from its return on day t:

$$A_{jt} = R_{jt} - \overline{R_j}.$$

The definitions of the average abnormal return and cumulative average abnormal return follow those for market model abnormal returns above.

A.2 Event Study Test Statistics

Standardized Abnormal Return Method

Eventus uses the standardized abnormal return method only for market model abnormal returns unless you specify `STDALL` on the `EVTSTUDY` statement.

Event Studies Centered on a Single Date

Eventus test statistics using standardized abnormal returns follow Patell (1976). Many published studies use the Patell test (see, for example, Linn and McConnell, 1983; Schipper and Smith, 1986; and Haw, Pastena and Lilien, 1990.)

Under the null hypothesis, each A_{jt} has mean zero and variance $\sigma^2_{A_{jt}}$. The maximum likelihood estimate of the variance is,

$$s^2_{A_{jt}} = s^2_{A_j}\left[1 + \frac{1}{D_j} + \frac{(R_{mt} - \overline{R_m})^2}{\sum_{k=T_{D_b}}^{T_{D_e}}(R_{mk} - \overline{R_m})^2}\right]$$

where

$$s^2_{A_j} = \frac{\sum_{k=T_{D_b}}^{T_{D_e}} A^2_{jk}}{D_j - 2},$$

R_{mt} is the observed return on the market index on day t, $\overline{R_m}$ is the mean market return over the estimation period and D_j is the number of non-missing trading day returns in the D-day interval T_{D_b} through T_{D_e} used to estimate the parameters for firm j.

Define the *standardized abnormal return* (or *standardized prediction error*) as

$$SAR_{jt} = \frac{A_{jt}}{s_{A_{jt}}}.$$

Under the null hypothesis, each SAR_{jt} follows a Student's t distribution with $D_j - 2$ degrees of freedom. Summing the SAR_{jt} across the sample, we obtain

$$TSAR_t = \sum_{j=1}^{N} SAR_{jt}.$$

The expected value of $TSAR_t$ is zero. The variance of $TSAR_t$ is

$$Q_t = \sum_{j=1}^{N} \frac{D_j - 2}{D_j - 4}.$$

The test statistic for the null hypothesis that $CAAR_{T_1,T_2} = 0$ is

$$Z_{T_1,T_2} = \frac{1}{\sqrt{N}} \sum_{j=1}^{N} Z^j_{T_1,T_2},$$

where

$$Z^j_{T_1,T_2} = \frac{1}{\sqrt{Q^j_{T_1,T_2}}} \sum_{t=T_1}^{T_2} SAR_{jt},$$

and

$$Q^j_{T_1,T_2} = (T_2 - T_1 + 1)\frac{D_j - 2}{D_j - 4}.$$

Under cross-sectional independence of the $Z^j_{T_1,T_2}$ and other conditions (see Patell, 1976), Z_{T_1,T_2} follows the standard normal distribution under the null hypothesis.

Eventus reports a precision-weighted cumulative average abnormal return with the standardized abnormal return method. The precision weighted average is constructed using the relative weights implied by the definition of Z_{T_1,T_2}. Thus, the precision weighted average will always have the same sign as the corresponding Z_{T_1,T_2}. The formula for the precision weighted average is

$$PWCAAR_{T_1,T_2} = \sum_{j=1}^{N} \sum_{t=T_1}^{T_2} w_j A_{jt},$$

where

$$w_j = \frac{\left(\sum_{t=T_1}^{T_2} s_{A_{jt}}^2\right)^{-\frac{1}{2}}}{\sum_{i=1}^{N}\left(\sum_{t=T_1}^{T_2} s_{A_{it}}^2\right)^{-\frac{1}{2}}}.$$

`TWIN` Event Studies (Two Firm-Specific Event Dates)

The major difference between `TWIN` and single date event studies is that `TWIN` cumulates returns over intervals of security-specific length. Instead of defining a window for return cumulation with reference to a single event date, the window is defined as the period between two event dates. The number of trading days between the two event dates varies from firm to firm.

Let the cumulative abnormal return for firm j be

$$CAR_{T_{1j},T_{2j}} = \sum_{t=T_{1j}}^{T_{2j}} A_{jt},$$

where T_{1j}, T_{2j} are the two event dates specific to firm j. Let L_j be the length of the event period in trading days,

$$L_j = T_{2j} - T_{1j} + 1.$$

The z statistic for testing the significance of $CAR_{T_{1j},T_{2j}}$ is

$$z_j = \frac{\sum_{t=T_{1j}}^{T_{2j}} SAR_{jt}}{(L_j \frac{D_j-2}{D_j-4})^{\frac{1}{2}}}.$$

Assuming cross-sectional and time-series independence, the test statistic for

$$CAAR = \sum_{j=1}^{N} CAR_{T_{1j},T_{2j}}$$

is

$$z_{CAAR} = N^{\frac{1}{2}} \sum_{j=1}^{N} z_j.$$

Correction for Correlation of Abnormal Returns

When the `SERIAL` option appears on the `EVTSTUDY` statement, Eventus uses a corrected version of the Patell test. The correction affects only windows, not single-period test statistics. Following Mikkelson and Partch (1988), the corrected test statistic for the null hypothesis that $CAAR = 0$ is

$$z_{CAAR} = N^{-\frac{1}{2}} \sum_{j=1}^{N} \frac{CAR_{T_{1j},T_{2j}}}{s_{CAR_{T_{1j},T_{2j}}}},$$

where

$$s^2_{CAR_{T_{1j},T_{2j}}} = s^2_{A_j} \left\{ L_j \left[1 + \frac{L_j}{D_j} + \frac{\left(\sum_{t=T_{1j}}^{T_{2j}} R_{mt} - L_j \overline{R_m} \right)^2}{\sum_{k=1}^{D_j} \left(R_{mk} - \overline{R_m} \right)^2} \right] \right\}. \tag{A.1}$$

For an event study centered on a single event date, T_{1j}, T_{2j} and L_j are equal across firms and the subscript j can be dropped from them.

The corrected test accounts for the fact that within the window, the abnormal returns for each stock are serially correlated. The serial correlation occurs because all the abnormal returns are functions of the same market model intercept and slope estimators. Applications of the corrected test in addition to Mikkelson and Partch (1988) include Mais, Moore and Rogers (1989), Cowan, Nayar and Singh (1990), Mann and Sicherman (1991) and Lee (1992). Simulation evidence of the properties of the corrected and uncorrected Patell tests appears in Karafiath and Spencer (1991, using Monte Carlo experiments) and Cowan (1993, using sampling experiments with CRSP data.) Both papers report that the bias in the uncorrected test is small in event windows shorter than 60 days but serious in event windows longer than 100 days. Mikkelson and Partch (1988) acknowledge Craig Ansley for the original derivation of the corrected test statistic in an event study context. For other derivations and discussion, see Cantrell, Maloney and Mitchell (1989) and Sweeney (1991).

When the SERIAL and STDALL options both appear on the EVTSTUDY statement, Eventus uses the following definitions for the standardized method tests with non-market model abnormal returns. For comparison period mean adjusted returns,

$$s^2_{CAR_{T_{1j},T_{2j}}} = s^2_{A_j}\left(L_j + \frac{L_j^2}{D_j}\right).$$

For raw returns and market-adjusted returns, there is no estimation of the mean. Instead, the mean is assumed to be equal to a known constant with probability one. The constant is zero in the case of raw returns and the realized market index return in the case of market-adjusted returns. Thus,

$$s^2_{CAR_{T_{1j},T_{2j}}} = s^2_{A_j}\left(L_j\right).$$

Standardized Cross-Sectional Method

Eventus uses the standardized cross-sectional method for market model abnormal returns when you specify STDCSECT on the EVTSTUDY statement. Boehmer, Musumeci and Poulsen (1991) introduce the method and document its empirical properties. The method is the same as the Patell standardized method described above except that there is a final empirical cross-sectional variance adjustment in place of the analytical variance of the total standardized prediction error. For additional discussion of event-date variance increases and related tests, see Sanders and Robins (1991).

For day t in the event period, the test statistic is

$$z_t = \frac{TSAR_t}{N^{\frac{1}{2}}\left(s_{SAR_{\bullet t}}\right)},$$

where

$$s^2_{SAR_{\bullet t}} = \frac{1}{N-1}\sum_{i=1}^{N}\left(SAR_{it} - \frac{1}{N}\sum_{j=1}^{N} SAR_{jt}\right)^2.$$

Eventus extends the cross-sectional standardized method to multiperiod windows using the correction for serial correlation described above. Thus, the STDCSECT option implies the SERIAL option. Define the standardized cumulative abnormal return for stock j as

$$SCAR_{T_{1j},T_{2j}} = \left(CAR_{T_{1j},T_{2j}} / s_{CAR_{T_{1j},T_{2j}}}\right),$$

where $s_{CAR_{T_{1j},T_{2j}}}$ is as defined in equation A.1. Then the standardized cross-sectional test statistic for the null hypothesis that $CAAR = 0$ is

$$z_t = \frac{\sum_{i=1}^{N} SCAR_{T_{1j},T_{2j}}}{N^{\frac{1}{2}} s_{SCAR_{\bullet t}}},$$

where

$$s^2_{SCAR_{\bullet t}} = \frac{1}{N-1} \sum_{i=1}^{N} \left(SCAR_{T_{1i},T_{2i}} - \frac{1}{N} \sum_{j=1}^{N} SCAR_{T_{1j},T_{2j}} \right)^2.$$

Time Series Standard Deviation Method

By default, the time series standard deviation method is applied to all benchmark methods (raw returns, comparison period mean adjusted, market adjusted, and market model) being estimated, to produce the results *not* labeled "standardized." Unlike the standardized abnormal return method, the time series standard deviation method calculates a single variance estimate for the entire portfolio. Therefore, the time series standard deviation method does not take account of unequal return variances across securities. On the other hand, it does avoid the potential problem of cross-sectional correlation of security returns. The estimated variance of AAR_t is

$$\hat{\sigma}^2_{AAR} = \frac{\sum_{t=T_{D_b}}^{T_{D_e}} (AAR_t - \overline{AAR})^2}{D-2},$$

where the market model parameters have been estimated over the estimation period of $D = T_{D_e} - T_{D_e} + 1$ days and

$$\overline{AAR} = \frac{\sum_{t=T_{D_b}}^{T_{D_e}} AAR_t}{D}.$$

The portfolio test statistic for day t in event time is

$$\mathrm{t} = \frac{AAR_t}{\hat{\sigma}_{AAR}}.$$

Assuming time-series independence, the test statistic for $CAAR_{T_1,T_2}$ is

$$\mathrm{t} = \frac{CAAR_t}{(T_2 - T_1 + 1)^{\frac{1}{2}}\,\hat{\sigma}_{AAR}}.$$

Many studies use the time series standard deviation method [see for example Dopuch, Holthausen and Leftwich (1986) and Brickley, Dark and Weisbach (1991)].

Cross-Sectional Standard Deviation Method

When the `CSECTERR` option appears on the `EVTSTUDY` statement, Eventus substitutes a daily cross-sectional standard deviation for the portfolio time-series standard deviation in the non-standardized tests. The portfolio test statistic for day t in event time is

$$\mathrm{t} = \frac{AAR_t}{\hat{\sigma}_{AAR_t}/\sqrt{N}},$$

where

$$\hat{\sigma}^2_{AAR_t} = \frac{1}{N-1}\sum_{i=1}^{N}\left(A_{it} - \frac{1}{N}\sum_{j=1}^{N} A_{jt}\right)^2.$$

The estimated variance of $CAAR_{T_1,T_2}$ is

$$\hat{\sigma}^2_{CAAR_{T_1,T_2}} = \frac{1}{N-1}\sum_{i=1}^{N}\left(CAR_{i,T_1,T_2} - \frac{1}{N}\sum_{j=1}^{N} CAR_{j,T_1,T_2}\right)^2.$$

The test statistic for $CAAR_{T_1,T_2}$ is

$$\mathrm{t}_{CAAR} = \frac{CAAR_{T_1,T_2}}{\hat{\sigma}_{CAAR_{T_1,T_2}}/\sqrt{N}}.$$

Brown and Warner (1985) report that the cross-sectional test is well-specified for event date variance increases but not very powerful. Boehmer, Musumeci and Poulsen (1991) report that the standardized cross-sectional test (see above) is more powerful and equally well specified. Cowan (1992) reports that the generalized sign test (see below) also is well specified for event date variance increases and more powerful than the cross-sectional test.

Generalized Sign Test

For each trading day or month in the event period, and for each window, Eventus reports the number of securities with positive and negative average abnormal returns (cumulative in the case of windows.) Also reported is a test statistic (in the default output format) and significance level symbols for the generalized sign test. The null hypothesis for the generalized sign test is that the fraction of positive returns is the same as in the estimation period. For example, if 46% of market adjusted returns are positive in the estimation period, while 60% of firms have positive market adjusted returns on event day -1, Eventus reports whether the difference between 60% and 46% is significant at the five percent, one percent, or one-tenth of one percent level. The actual test uses the normal approximation to the binomial distribution. For examples of the generalized sign test in recent research, see Sanger and Peterson (1990), Singh, Cowan and Nayar (1991), and Chen, Hu and Shieh (1991). (Chen, Hu and Shieh refer to the test as a binomial sign test.) For a more detailed explanation of the generalized sign test, see Sprent (1989) and Cowan (1992).[2]

Rank Test

Corrado (1989) describes the rank test for a one-day event window. The ranks of the abnormal returns of different days are dependent by construction. However, the effect of ignoring the dependence should be negligible for short event windows. Eventus extends the rank test to multiple day windows by assuming that the daily return ranks within the window are independent.

The rank test procedure treats the combined estimation period and event period as a single set of returns, and assigns a rank to each daily (or monthly, etc.) return for each firm.[3] Let K_{jt} represent the rank of abnormal return A_{jt} in the sample of $D_j + E_j$ abnormal returns of stock j. E_j is the number of non-missing returns of stock j in the event period; if there are no missing returns, $E_j = E =$ `POST` $-$ `PRE` $+ 1$ and $D_j = D =$ `ESTLEN`. Rank one signifies the smallest abnormal return. The mean (and median) rank across

[2] Corrado and Zivney (1992) analyze a similar sign test.

[3] Eventus does not require that the estimation and event periods be contiguous. The estimation and event period returns used for the other computations in the current event study are used for the rank test also.

the combined estimation and event period is

$$\widetilde{K} = \frac{D+E+1}{2}.$$

The rank test statistic for the event window composed of days T_1 through T_2 is

$$z_r = (T_2 - T_1 + 1)^{\frac{1}{2}} \left\{ \frac{\overline{K_{T_1,T_2}} - \widetilde{K}}{\left[\sum_{t=1}^{D+E} \left(\overline{K_t} - \widetilde{K} \right)^2 / (D+E) \right]^{\frac{1}{2}}} \right\},$$

where

$$\overline{K_{T_1,T_2}} = \frac{1}{T_2 - T_1 + 1} \sum_{t=T_1}^{T_2} \frac{1}{n} \sum_{j=1}^{n} K_{jt}$$

is the average rank across the n stocks and $T_2 - T_1 + 1$ days of the event window and $\overline{K_t} = (1/n) \sum_{j=1}^{n} K_{jt}$ is the average rank across n stocks on day t of the $D+E$ day combined estimation and event period. The expected rank still is $\widetilde{K}$ for event windows shorter than E days, because the full $D+E$ day set of returns is used for the assignment of ranks.

Jackknife Test

The discussion in this subsection is adapted from Giaccotto and Sfiridis (1996). The jackknife test incorporates the standardized abnormal return for each stock j, computed using the event period sample standard deviation. The standardized abnormal return for day t is

$$\hat{\theta} = \frac{A_{jt}}{\tilde{\sigma}_{A_{jt}}} \tag{A.2}$$

where

$$\tilde{\sigma}_{A_{jt}} = \left\{ \sum_{t=T_b}^{T_e} \frac{\left(A_{jt} - \overline{A_j}^2 \right)}{E_j} \right\}^{\frac{1}{2}} \tag{A.3}$$

and $\overline{A_j}$ is the mean abnormal return of stock j during the event period of $E = T_e - T_b + 1$ days. If there is an event-induced, transient variance change on day t, then $\tilde{\sigma}_{A_{jt}}$ is a biased estimator of $\sigma_{A_{jt}}$ and $\hat{\theta}$ is a biased statistic. Giaccotto and Sfiridis propose reducing the bias by jackknifing the $\hat{\theta}$ values.

The first step of the jackknife is to sequentially delete one abnormal return A_{jT_s} from equation A.3 and re-compute $\tilde{\sigma}_{A_{jt}}$, using the new value in turn to re-compute $\hat{\theta}$ using equation A.2. Call the latter value $\hat{\theta}_{(-s)}$. The next step is to form pseudo-values

$$\theta_{(-s)} = (E_j)\,\hat{\theta} - (E_j - 1)\,\hat{\theta}_{(-s)}$$

The jackknife estimator for stock j on day t is the mean of the pseudo-values,

$$\theta_{jt} = \frac{1}{E_j} \sum_{t=T_b}^{T_e} \theta_{(-s)}$$

To gain efficiency, the estimates are averaged across the sample of stocks:

$$\overline{\Theta}_t = \frac{1}{N} \sum_{j=1}^{N} \theta_{jt}$$

Finally, the jackknife test statistic for the sample of stocks on day t is

$$t_{Jackknife} = \frac{\overline{\Theta}_t}{S_{Jackknife,t}/\sqrt{N}} \tag{A.4}$$

where

$$S_{Jackknife,t} = \left[\frac{1}{N-1} \sum_{i=1}^{N} \left(\theta_{jt} - \overline{\Theta}_t \right)^2 \right]^{\frac{1}{2}}. \tag{A.5}$$

The distribution of $t_{Jackknife}$ under the null hypothesis is approximately normal with mean zero and unit variance.

To test the significance of the cumulative average abnormal return over the window from date T_1 through date T_2, define

$$\hat{\theta}_{T_1,T_2} = \frac{\sum_{t=T_1}^{T_2} A_{jt}}{(T_2 - T_1 + 1)^{\frac{1}{2}}\,\tilde{\sigma}_{A_{jt}}} \tag{A.6}$$

Sequentially delete one abnormal return A_{jT_s} from equation A.3 and re-compute $\tilde{\sigma}_{A_{jt}}$, using the new value in turn to re-compute $\hat{\theta}$ using equation A.6. Call the latter value $\hat{\theta}_{(-s),T_1,T_2}$. Form pseudo-values

$$\theta_{(-s),T_1,T_2} = (E_j)\,\hat{\theta}_{T_1,T_2} - (E_j - 1)\,\hat{\theta}_{(-s),T_1,T_2}$$

The jackknife estimator for stock j in window (T_1, T_2) is the mean of the pseudo-values,

$$\theta_{j,T_1,T_2} = \frac{1}{E_j} \sum_{t=E_b}^{E_e} \theta_{(-s)}$$

The estimates are averaged across the sample of stocks:

$$\overline{\Theta}_{T_1,T_2} = \frac{1}{N} \sum_{j=1}^{N} \theta_{j,T_1,T_2}$$

The jackknife test statistic for the sample of stocks in window (T_1, T_2) is

$$t_{Jackknife} = \frac{\overline{\Theta}_{T_1,T_2}}{S_{Jackknife,T_1,T_2}/\sqrt{N}} \tag{A.7}$$

where

$$S_{Jackknife,T_1,T_2} = \left[\frac{1}{N-1} \sum_{i=1}^{N} \left(\theta_{j,T_1,T_2} - \overline{\Theta}_{T_1,T_2} \right)^2 \right]^{\frac{1}{2}}. \tag{A.8}$$

A.3 Different estimation and event return intervals

The `ESTINTER` option on the `EVENTUS` statement allows the use of different return intervals for parameter estimation and event testing. For example, it is possible to conduct a daily-return event study but estimate the market model slope, intercept and standard error from monthly returns. Eventus adjusts the estimated intercept and standard error for the difference in the return interval. The market model intercept, mean estimation period market index return, and comparison period mean stock return are divided by the adjustment factor shown in table A.1; the standard error is divided by the square root of the adjustment factor. The number of returns in the estimation period, D_j, is multiplied by the adjustment factor. Other reported statistics, such as the estimation period standard deviation of raw returns, reflect appropriate adjustments using the same factor.

Table A.1
Adjustments factors for different estimation and event return intervals.

Event Period	Estimation Period Interval				
	Day	Week	Month	Quarter	Year
Day	1.00	4.88	21.08	63.25	253.00
Week	0.21	1.00	4.32	12.96	51.84
Month	0.05	0.23	1.00	3.00	12.00
Quarter	0.02	0.08	0.33	1.00	4.00
Year	0.01	0.02	0.08	0.25	1.00

Note: Values are rounded to two decimal places for display. Internally, values for event period intervals longer than a day are calculated from the values for daily event periods and are not rounded.

A.4 Variable Names In Eventus Output SAS Data Sets

Users who want to write their own programs to process event study data saved by Eventus need to know the variable names used.

Data sets created with the `OUTSAS` option on the `EVTSTUDY` statement represent each day by one or more variables. Any variable ending in the number 255 represents the last day of the estimation period. The first estimation period variable in a series depends on the length of the estimation period. If the estimation period is 100 trading days long, for example, the first day is represented by a variable ending in the number 156. The sequence of variable numbers is in ascending time. Thus, variable number 255 contains a data item for day −46 if `EST=-46` is specified on the `REQUEST` statement. If `EST=+91 ESTLEN=100` were specified, variable number 255 would represent day +190.

Variable names ending in the numeral 256 represent the first day of the event period, regardless of estimation period length. For example, if `PRE=20` is specified, then variable number 256 corresponds to day −20.

Table A.2 presents the possible range of variable names in an output event study data set (`OUTSAS` data set.) Only those variables needed to represent days as described above are included in an actual data set.

`_WEIGHT_` is a five-byte character variable. It will have a value of `'Equal'`, `'Value'`, `'SP500'`, `'NASDQ'` depending on the market index used. When both the equally weighted index and another index are used, there are two

Table A.2
Variable names in a saved event study data set.

Data	Variable Name	
	Estimation Period	Event Period
Daily abnormal return	AR1-AR255	AR256-AR9999
Daily abnormal return[a]	ARX1-AR744	
Daily raw return	RETN1-RETN255	RETN256-RETN9999
Daily raw return[a]	RETNX1-RETNX744	
Daily market index return		
Equally weighted	MKT1-MKT255	MKT256-MKT9999
Equally weighted[a]	MKTX1-MKTX744	
Value weighted	VWMK1-VWMK255	VWMK256-VWMK9999
Value weighted[a]	VWMKX1-VWMKX744	
$s_{A_{jt}}^{-1}$	—	WDEN256-WDEN9999
Dummy variable =1 if abnormal return> 0	—	NPOS256-NPOS9999
Parameters		
$\hat{\alpha}_j$	ALPHA	
$\hat{\beta}_j$	BETA	
mean return	CPR	
$\overline{R_m}$	RMBAR	
$\sum_{k=T_{D_b}}^{T_{D_e}}(R_{mk}-\overline{R_m})^2$	SSRM	
return variance	OWNVAR	
s_{A_j}	_RMSE_	
Fraction of returns > 0	NPOS_EST	
D_j	Ti	
Name of market index	_WEIGHT_	
Type of abnormal return	RESTYPE	

[a] Additional variable names, used only when the estimation period exceeds 255 trading days (or other intervals.) "X" variable subscripts start with 1 and go up to $T - 255$ where T is the length of the estimation period. The "X" variables contain the first T days of the estimation period, then subscript 1 of the regular variables continues with the $T + 1$st day. For example, if ESTLEN=300, the first 55 days of event period abnormal stock returns are in ARX1 through ARX45 and the 56th through 300th days are in AR1 through AR255.

stored observations per stock-event date combination per abnormal return method: one for each value of _WEIGHT_. When CRSP excess returns (based upon beta- or standard deviation ranks) are in use, _WEIGHT_ takes the value 'EXCES'.

The RESTYPE variable takes a three-character value of 'raw' for unadjusted raw returns, 'CP' for comparison period mean-adjusted, 'MAR' for market adjusted, 'MM' for OLS, GARCH or EGARCH market model adjusted, 'SW' for Scholes-Williams market model adjusted returns, or 'XS' for CRSP Excess Return File excess returns.

A.5 Missing Returns

CRSP codes any missing return on its files as an integer strictly less than -1.0. Eventus internally converts each of these CRSP missing return codes to the SAS special missing value. Special missing values work the same way as the regular SAS missing value, ., in all arithmetic operations.

When an estimation period contains a sequence of one or more missing values, Eventus does not use the first succeeding non-missing return. The reason is that the first non-missing return is a multi-period return. Permitting multi-period returns could have unexpected consequences for parameter estimates. The first non-missing return following a sequence of missing estimation period returns is replaced by the special missing value .N.

When a sequence of one or more returns is missing in the event period, Eventus adjusts the abnormal return computation procedure to account for the multi-day character of the first post-missing return. For example, if the number of missing days is q, the market model abnormal return for the first post-missing day, t, is

$$A_{jt} = R_{jt} - \left[(q+1)\hat{\alpha}_j + \hat{\beta}_j \sum_{h=0}^{q} R_{m(t-h)} \right],$$

while the maximum likelihood estimate of the variance of A_{jt} is,

$$s^2_{A_{jt}} = s^2_{A_j} \left\{ (q+1)(1 + \frac{1}{D_j}) + \frac{\left[\sum_{h=0}^{q} R_{m(t-h)} - (q+1)\overline{R_m} \right]^2}{\sum_{k=T_{D_e}}^{T_{D_b}} (R_{mk} - \overline{R_m})^2} \right\}.$$

Table A.3
Eventus special SAS missing values for missing returns from the CRSP database.

Reason	CRSP missing return code	Special SAS missing value
No portfolio assignment for excess-return file	−44	.X
Missing delisting price	−55	.D
> 10 trading days between non-missing prices	−66	.G
No trading on Nasdaq	−77	.T
Date outside return date range	−88	.R
No price available	−99	.B
Event dropped	NA	.M
Date outside Eventus search range	NA	.A
Day in estimation period after a missing-return day	NA	.N

Appendix B

Reference Guide to Eventus Statements

B.1 CUSIPERM Statement

This statement is used after the EVENTUS statement to convert CUSIP identifiers to PERMNOs. The statement reads the user request file and performs the conversion. The user may specify these options:

COLUMN=*n* Use when the CUSIP is not the first item on each line of the request file. Substitute the starting column for the CUSIP for *n*. This option is not needed if only blanks precede the CUSIP on each line.

EXTFILE= Gives the fileref of the external file in which Eventus is to store the updated copy of the request file. The default is `request`. On mainframe systems where applicable, the file should have a disposition of `NEW` or `OLD` and a logical record length of 80.

REQFILE=*fileref* The request file for `CUSIPERM` must be an external (non-SAS formatted) file, such as a card image format file. Replace *fileref* with the fileref associated with your file. The `REQFILE` specification may be omitted if the fileref of the request file is `REQUEST`. The file need not be sorted by CUSIP.

B.2 DATECONV Statement

The DATECONV statement is used after the EVENTUS statement to convert dates between calendar and CRSP formats. It reads the user request file and performs the date conversion operation. The user may specify these options:

AUTODATE Specifies that any calendar date in the request file that is not a trading day should be converted to the following trading day.

CUSIP Specifies that the CRSP database or USERSTOK file to be read is sorted by CUSIP and the request file contains CUSIPs instead of PERMNOs. This option is provided for compatibility with the REQUEST statement.

CUSIPERM Specifies that the request file contains CUSIPs but a CRSP database sorted by PERMNO is to be read. Eventus will attempt to convert CUSIPs to CRSP PERMNOs. The option requires a conversion table, a component of Eventus created by the PermnoUp program at Eventus installation and after each update of the CRSP database.

DATEFMT=*format* Specifies the input format of the dates being read from the request file. The specification must be either a valid SAS date informat or the word CRSP. The word CRSP tells Eventus to look for a 1–to–4 digit integer representing a CRSP trading day (or month) number (1=July 2, 1962 on the daily calendar, except 1=December 14, 1972 for early versions (1985 and 1987) of the Nasdaq database.) Leading zeroes need not (but may) be included in the CRSP day number. Any format other than CRSP must be a valid SAS date format (although the period at the end is optional.) The default, DATEFMT= YYMMDD, accommodates both two digit and four digit years automatically.

EST=*periods* **and POOL** These options are provided for compatibility with the REQUEST statement; they have no effect when specified on the DATECONV statement.

ESTLEN=*n* This option is provided for compatibility with the REQUEST statement.

EXTFILE=*fileref*|*file name* Gives the fileref or name of the external file in which Eventus is to store the converted request file. If the argument is a file name, it may include the path but must not include any blank

or period. On most systems, a `.dat` extension will be added to the file name automatically. The default is `USERDATA`. On mainframe systems where the file is pre-allocated by control language or in the `filename` statement, the file should be given a disposition of `NEW` or `OLD`, a fixed block format, and a logical record length of 80.

GROUP=*variable* Names a grouping variable to be used in an event study to combine multiple observations into a single equally weighted portfolio. The value of the grouping variable for each observation is listed on the appropriate line of the request file after the dates and `ID` variable (if any.) The grouping variable must be an integer between 0 and 9999 inclusive; leading zeros in the request file are optional and ignored. The grouping variable is included in the converted request file.

GRWEIGHT Valid only if the `GROUP` option is specified. Denotes that the request file contains a group weight variable. This variable, expressed as a decimal, specifies the weight to be given the individual stock within its group portfolio. All the weights for a single group should sum to 1. This option is included on the `DATECONV` statement for compatibility with the `REQUEST` statement; weights are copied to the output file.

ID=*variable* Optionally names a variable to be used as an observation identifier. The identifying variable may be of any data type. If `INSAS` is specified, the identifying variable must exist on the permanent SAS data set specified.

IDFMT=*format* Gives the format of the identifying variable in the external request file. For example, if the identifying variable is an integer that varies from one to four digits, specify `IDFMT=4.0`.

INSAS=*libref.membername* Used in place of `REQFILE` when the input request file is a SAS data set. The input SAS data set must contain these variables:

- `CUSIP` (8-character string.) Required if and only if `CUSIP` or `CUSIPERM` appears on the `DATECONV` statement.
- `PERMNO` (5-digit integer.) Required unless `CUSIP` or `CUSIPERM` appears on the `DATECONV` statement.

- The dates to be converted. For a single date per PERMNO, either `EVENTDAT` (SAS date variable) or `CRSPDAY` (integer representing trading date 0 as a sequence number corresponding to the CRSP index calendar. The integer may represent the day, week, month, quarter or year from the beginning date of the index file.) For programs that convert two dates per PERMNO (specify `TWIN` on the `EVENTUS` statement), two names must appear: `EVENTDA1` and `EVENTDA2`, or `CRSPDAY1` and `CRSPDAY2`. If you want Eventus to look for `CRSPDAY` (and `CRSPEST`), specify `DATEFMT=CRSP` (see below.)
- If you specify `ID=`*variable* (see above), the *variable* named there.

INSAS2=*libref.membername2* This option is used only with an SFA format CRSP database. Specifies a second request file in the form of a SAS data set. A second CRSP stock file, which must be associated with the `CRSPST2` libref, is searched for the PERMNOs in the second request file. The format required for the second request file is the same as for the first.

IX2Y Included for compatibility with the `REQUEST` statement only.

NAME Indicates that the input request file includes a firm name. Rarely used.

NDAYS=*n* Specifies that you want to create a second date by adding *n* trading days (months) to the first date for each PERMNO on each line of the request file. If you specify `NDAYS`, omit the second date from your request file.

OUTDTFMT=*format* Gives the date format for Eventus to use in creating the converted request file. Valid formats are all SAS date formats and `CRSP`. The default is `YYMMDD8.`.

OUTSAS= Gives the libref and member name of the SAS data set to create for the converted request file. This option should be used only when the converted request file is desired in a SAS data set format, as opposed to an ASCII or EBCDIC text file. The latter is created by the `EXTFILE` option or default. Do not specify both `EXTFILE` and `OUTSAS`.

REQFILE=*fileref* The most often used method of storing the request information (PERMNO identifiers, dates, etc.) is to use an external (non-SAS formatted) file, typically an ASCII text file. Replace *fileref* with the fileref associated with your file. The `REQFILE` specification may be omitted if the fileref of the request file is `REQUEST`. Each line of the request file should contain the following variables in order:

```
PERMNO event  [event       [specific     [ID number  [grouping  [group  [S or
        date   date 2]  estimation date]  or string]  variable]  weight]  L]
```

Each value must be separated by at least one blank, but the exact position of the values is unimportant as long as they appear in the order shown. The square brackets simply indicate items that need not always appear; do not include them in the file. Whether an optional item should appear is determined by the options specified on the `EVENTUS` and `DATECONV` statements. The file need not be sorted by PERMNO. If the `CUSIPERM` option or the `CUSIP` option is specified, the first variable on the line should be `CUSIP` instead of `PERMNO`.

REQFILE2=*fileref* Specifies a second request file to be used in the same job. The format required for the second request file is the same as for the first.

SHIFT1=n_1,SHIFT2=n_2 The first date in the request file is shifted by n_1 periods and the second date is shifted by n_2 periods. For the monthly file, the periods are months. For the daily file, the periods are trading days if `DATEFMT=CRSP` and calendar days otherwise. Both n_1 and n_2 may be specified as any integer value. For example, `SHIFT1=-1` shifts June 1, 2000 back to May 31, 2000.

`SHIFT1` and `SHIFT2` may be specified singly as well as together. Using these options with calendar dates may result in invalid date messages unless `AUTODATE` is also specified.

SHORT This option is provided for compatibility with the `REQUEST` statement. The request file must contain an `S` or `L` code, which is copied to the converted request file, at the end of each line.

SORTBYID Specifies that the converted request file is to be sorted by the variable listed in the `ID` option. The default is to sort by PERMNO unless the `CUSIP` option is specified, in which case the default is to sort by CUSIP.

UPCUSIP This option is valid only when `CUSIP` also is in effect. Specifies to try to update CUSIP identifiers in the request file to match the latest version of the CRSP stock file. When a change is made, only the updated CUSIPs appear in the converted request file. The option requires a conversion table, a component of Eventus created by the PermnoUp program at Eventus installation and after each update of the CRSP database. This option is nearly obsolete now that CRSP stock files are sorted by PERMNO.

B.3 `EVENTUS` statement

The `EVENTUS` statement is always required. It has the following options.

ACCESS97[=0] Indicates that the input CRSP database is CRSPAccess format. Specifying `=0` turns the option off. Normally set at Eventus installation but may be specified at run time.

ANNUAL Tells Eventus that it is reading annual returns from a `USERSTOK` file (also see the `NONCRSP` option) or from an annual returns database formatted like CRSP stock databases.

CHAR[=0],CHAR4[=0] `CHAR` indicates that the input CRSP database is SFA character format. `CHAR4` further specifies that the SFA character format database contains four digit years. Specifying `=0` turns the option off. Normally set at Eventus installation but may be specified at run time.

DBFNSTMT[=0] Indicates that input CRSPAccess format database is identified by a SAS `filename crspdb` statement instead of by environment variables or logicals. Specifying `=0` turns the option off. Normally set at Eventus installation but may be specified at run time.

DUAL Available for event studies only, the `DUAL` option indicates that there are two CRSP SFA format stock files. The filerefs of the files must be `CRSPSTOK` and `CRSPST2`. The user must supply two request files corresponding to the two stock files. `DUAL` is the same as `REQFILES=2`. This option is obsolescent because of the ubiquity of merged NYSE-AMEX-Nasdaq data. The `DUAL` option is not supported for CRSPAccess format nor for CRSP data installed in SAS data sets.

ELIBNAME=*libref* This option is only used in situations when both the ESTINTER and LIBNAME options also are specified. Gives the SAS libref of the CRSP stock files to be searched for the estimation period return data. This option is for systems that store the CRSP data in the form of SAS data sets. In the ELIBNAME library, Eventus expects to find SAS data sets named: cal (the CRSP stock index file for the estimation period), header, and tsdata.

ESTINTER=*interval* The holding interval of the estimation period returns, if different from the holding interval of event period returns. Valid values of *interval* are YEAR, QUARTER, MONTH, WEEK, and DAY.

EXCESS When working with an SFA format CRSP database, specifies that the CRSP excess returns file is associated with the fileref CRSPSTOK. When working with a CRSPAccess format database, specifies that Eventus should substitute a beta-matched (standard deviation-matched if SPORT is also specified) portfolio return for the market index return. Requires an add-on CRSP data subscription.

FBIN[=0|2] When your site receives the Eventus installation files, the default for this option is set according to the type of CRSP files you have. On some systems, a fixed-length binary format file formerly had to be used instead of the variable-length format generated from a character format file by CRSP subroutines. FBIN specifies a fixed-length file in native binary format; FBIN=2 specifies a fixed-length file in IBM binary format. Specify this option in an Eventus program only when the type of file you want to use is not the default for your site. This option is now obsolete.

GETDATA Specify this option only in a program that uses the RETURNS or PRICES statement to read and store CRSP stock data.

HOSTBIN[=0] Indicates that the input CRSP database is SFA binary format with either two digit or four digit years. Specifying =0 turns the option off. Normally set at Eventus installation but may be specified at run time.

IDXLEAD=*n* Indicates the number of bytes to skip at the beginning of each record of an input SFA binary CRSP calendar/index file to compensate

for record length blocks. Defaults to zero and ignored with CRSPAccess format and non-CRSP data. Normally set at Eventus installation but may be specified at run time for experimentation with alternative values.

LIBNAME=*libref* Gives the SAS libref of the CRSP stock data sets. This option is for systems that store the CRSP data in the form of SAS data sets. Eventus expects to find SAS data sets named: `cal` (the CRSP stock index file); `header`, `names`, `shares`, `nasdin`, `dists`, `prtnum` (if a size-decile index is to be used), and `tsdata`.

MONTHLY Indicates that Eventus is to read from the monthly CRSPAccess database, or that the appropriate fileref or libref already points to a CRSP SFA database, CRSP data stored in SAS data sets, or `USERSTOK` file that contains monthly data. When working with CRSP SFA (old) format data files, a `filename` statement, environment variable or control language must associate the fileref `CRSPSTOK` (and `CRSPST2` etc. when applicable) with the proper monthly file if you use this option.

NASDAQ The `NASDAQ` option tells Eventus that the fileref `CRSPSTOK` is associated with an old Nasdaq file instead of an old NYSE-AMEX file. A `filename` statement, environment variable, logical variable or control language must associate the fileref `CRSPSTOK` with the old CRSP Nasdaq database if you use this option for an event study or to pull returns, prices or volume data. The `NASDAQ` option is not supported for CRSPAccess format nor for CRSP data installed in SAS data sets.

NLIBNAME=*libref* Gives the SAS libref of the CRSP Supplemental Nasdaq data file. This option is for systems that store the CRSP data in the form of SAS data sets, and is needed only in runs where the `NMS` option appears on the `PRICES` statement. In the `NLIBNAME` library, Eventus expects to find SAS data sets named `nmshdr` and `nmsdata`.

NONCRSP Denotes that user-extracted stock and index return data are to be used. Eventus does not try to search a CRSP database when `NONCRSP` is in effect.

PAGE=TALL|WIDE The default is `PAGE=TALL`, which causes printed output to be formatted with the SAS system option `LS=79` for portrait orientation. `PAGE=WIDE` causes output to be formatted with the SAS

system option LS=132 for landscape orientation; this requires a small font or a wide-carriage printer.

PRE95 Specifies that the ending date of the input CRSP database is before 1995. This is not needed and has no effect with SFA character format CRSP database.

QUARTER This tells Eventus that it is reading quarterly returns from a USERSTOK file (also see the NONCRSP option) or from a quarterly returns database formatted like CRSP stock databases.

REQFILES=*n* An extension of the DUAL option for event studies, this option indicates that there is more than one CRSP stock file, with a request file corresponding to each. The filerefs of the files must be CRSPSTOK, CRSPST2, ... CRSPST*n*. The user must supply corresponding request files with filerefs REQUEST, REQUEST2, ..., REQUEST*n*. (If *n*=2, the request files can be SAS data sets; in this case, the filerefs would not be used.) The value of *n* may be from 2 to 9. If neither DUAL nor REQFILES is specified, Eventus looks for only one request file and one CRSP stock file.

SASCRSP[=0] Indicates that the CRSP data are in the form of SAS data sets. Specifying =0 turns the option off. Normally set at Eventus installation but can be specified at run time.

SIZEIND=*libref.membername* Specifies the two-level name of the SAS data set containing the size-decile return data. This option is for systems that store the CRSP data in the form of SAS data sets, and is needed only in jobs where the REQUEST statement specifies the SIZEINDX option. Eventus expects to find variables named caldt and decret1 through decret10 in the SIZEIND data set.

TWIN Specify TWIN in event study programs when you want to estimate variable-length (firm-specific) event windows, instead of the conventional abnormal returns and windows around a single firm-specific event date. When TWIN is in effect, you provide two event dates for each observation instead of just one. See the REQUEST statement description below for an explanation of how to indicate the event dates. TWIN is also used this way in DATECONV programs to convert a pair of dates instead of just one.

VAXFMT Causes Eventus to use the `VAXRB4.` SAS informat instead of the default `FLOAT4.` SAS informat to read an input CRSP database that is SFA binary or CRSPAccess format. This option is intended primarily for OpenVMS/Alpha systems. It has no effect when the input CRSP database is SFA character format.

WEEKLY Tells Eventus that it is reading weekly returns from a `USERSTOK` file (also see the `NONCRSP` option) or from a weekly returns database formatted like CRSP stock databases.

B.4 `EVTSTUDY` Statement

`EVTSTUDY` is required to run an event study.

ALLDAYS If the number of days covered by the event study, `PRE+POST+1`, exceeds 98, not all the individual day portfolio returns are listed by default. Day 0 is always listed, together with a proportionate number of contiguous `PRE` and `POST` days. All the windows you request are printed, regardless of their date ranges. To obtain a complete printout of all the individual days' portfolio returns, specify `ALLDAYS`. This option is ignored for `TWIN` event studies.

BOOT Requests bootstrap tests for window abnormal returns. Bootstrap tests are in addition to the normally reported parametric and nonparametric tests. `BOOT` implies the `BUYHOLD` and `STDCSECT` options. It is not possible in the current version of Eventus to bootstrap `TWIN` event studies.

BOTH Please see `VALUE`.

BTAIL=1|2 Selects one- or two-tailed bootstrap tests. The default is 2. This option is ignored unless the `BOOT` option also is specified.

BUYHOLD Specifies buy-and-hold compounded return computation for windows instead of the default additive cumulation.

CDCSI Requests the Collins and Dent (1984) test assuming cross-sectional independence (Sanders and Robins, 1991) instead of the default standardized test.

CP Requests comparison-period mean-adjusted returns. Does not affect market adjusted and market model returns.

CSECTERR Substitutes a cross-sectional standard deviation for the default time-series standard deviation in non-standardized t-statistic computations.

DETAIL Causes Eventus to report the windows for each firm individually under the standardized abnormal return method, in addition to the usual portfolio statistics.

DETAIL=FULL This option gives you the output described under `DETAIL`, plus the individual daily returns for each observation. Warning: this option produces at least one page and over 50 lines of printed output for each firm.

EGARCH Causes the market model to be estimated assuming an exponential GARCH(1,1) error process. The `EGARCH, GARCH` and `SW` options are mutually exclusive.

EGLS Requests the estimated generalized least squares test (Sanders and Robins, 1991) instead of the default standardized test.

FACTORS=*n* Specifies how many exogenous factors are to be used for the `OLSPARAM`, `SUR`, or `ITSUR` option. The default is `FACTORS=1`, indicating that only one factor (typically the market index) is to be used. Non-default values of this option are allowed only when the `NONCRSP` option appears on the `EVENTUS` statement.

GARCH Causes the market model to be estimated assuming a GARCH(1,1) error process. The `EGARCH, GARCH` and `SW` options are mutually exclusive.

INSAS=*libref.membername* Points to the SAS data set from which to take the user-supplied return information when the `NONCRSP` option appears on the `EVENTUS` statement. The file must include the following variable names: `CUSIP, RETURN, MARKET` and the variable specified in the `ID` option, if any. While the variable name `CUSIP` is required, the variable itself can contain any character string up to eight characters in length, for example a Datastream mnemonic code. For each observation in

the request file, there must be ESTLEN+PRE+POST+1 observations in the INSAS file. The default if the INSAS option is not specified is to read the user-supplied data from an external file associated with the fileref USERSTOK instead of a SAS data set. When the INSAS option is in effect, there is not need to define the fileref USERSTOK. In this case, the input SAS data set is called the "USERSTOK file" even though it is not a plain text file.

ITSUR Causes Eventus to use the iterated seemingly unrelated regressions approach, where the market model, augmented by a dummy variable for each window specified on the WINDOWS statement, is estimated by iterated joint generalized least squares over the combined estimation period and event period return series.

JACKNIFE Designates the jackknife test instead of the generalized sign test as the nonparametric test to accompany non-standardized method parametric tests.

MAXMISS=*n* Specifies to remove from the sample any firm-event with more than *n* days or months of missing returns in the event period.

MEDIAN Selects printing of the median abnormal return in place of the number positive and negative and nonparametric significance level, except in the standardized method. The option is ignored when PAGE=TALL is in effect, because both the median and the number positive and negative are always printed in this situation. Affects printed output only and affects only the daily or monthly output, not the output for windows.

NONAMES Suppresses printing of the list of PERMNOs or CUSIPs, firm names, and number of estimation and event period returns found. The default is to print the list.

NOPLIST Prevents the printing of the estimation period statistics that normally appear between the input report and the event study results. NOPRINT implies NOPLIST.

NOPRINT A report of the CRSP database input operation is produced, but all other event study printed output is suppressed. An output data set still is created if the OUTSAS option is specified. This option was

designed primarily for a batch environment and can be unstable in an interactive SAS session. The NONAMES and NOPLIST options can be combined to reduce output in an interactive setting.

NUMFM=*SAS_numeric_informat* Specifies the numeric format of stock and market index returns in an input USERSTOK file for a non-CRSP event study. Must be a valid SAS numeric informat. Defaults to 11.6 (11 character positions wide including sign, decimal point and 6 digits to the right of the decimal point.)

OLSPARAM Selects the event parameter approach, where the market model, augmented by a dummy variable for each window specified on the WINDOWS statement, is estimated on the combined estimation period and event period return series.

OUTSAS=*libref.membername* Specifies a SAS data set in which to save abnormal returns and other data for each event. Eventus will create the SAS data set named if it does not already exist, or replace it if it does. The use of a different second level name for each data set permits a single SAS data library to contain multiple Eventus event study data sets. See the PACKAGE= option below for a detailed description of the contents of the event study data set.

OVERLAP Suppresses checking for overlapping estimation period and event period.

PACKAGE=*specifier*[*specifier...*] Determines the content of the SAS data set created when OUTSAS is specified. Table B.1 lists the available specifiers. The OUTSAS file always contains at least the variables PERMNO, NAME, identifying variable if any, CRSPDAY and EVENTDAT (or CRSPDAY1, CRSPDAY2, EVENTDA1 and EVENTDA2 for TWIN event studies), _WEIGHT_ and RESTYPE. When the CUSIPERM option appears on the REQUEST statement, the saved data set includes the CUSIP variable in addition to PERMNO. When the SIZEINDX option appears on the REQUEST statement, the saved data set includes the variable __CAP__, the size-based portfolio number. CRSP assigns the number 1 to the smallest market capitalization portfolio and 10 to the largest. Note that a stock's size portfolio membership changes from year to year and can differ between the now standard NYSE-AMEX-Nasdaq database and the formerly common exchange-specific CRSP files. The default is PACKAGE=1.

Table B.1
PACKAGE specifiers for the EVTSTUDY statement.

Data[1]	Package Specifier		Remarks
	Estimation Period	Event Period	
Cumulative abnormal return		0	TWIN only.
Data required for OLDSTUDY		1	Same as OADGHP.
Daily abnormal returns	A	D	
Daily raw returns	B	E	Even if RAW not specified.[2]
Daily market index returns	C	F	
Daily standard deviation		G	Ignored if NOSTD specified.
Dummy variable =1 if abnormal return> 0		H	Mainly for OLDSTUDY use.
Parameters	P		Mean returns, α, β, etc.

[1]Table A.2 on page 117 lists SAS variable names. You do not need to know variable names to use the EXTRACT or OLDSTUDY statements.
[2]The EXTRACT statement will not extract raw returns unless the EVTSTUDY statement that produced the SAS data set included RAW and PACKAGE=D....

POST=*periods* Specifies the number of trading days or months immediately following the event date for which to compute abnormal returns. The default is POST=30 for daily and weekly and POST=12 for monthly event studies.

PRE=*periods* Specifies the number of trading days or months immediately preceding the event date for which to compute abnormal returns. The default is PRE=30 for daily and weekly event studies and PRE=12 for monthly event studies.

RANKTEST Designates the rank test, instead of the generalized sign test, as the nonparametric statistic to appear with the non-standardized parametric tests.

SERIAL Specifies that the window z tests should be corrected for the serial correlation of abnormal returns that is present by construction. This option is implied by the STDCSECT option.

SHRCODE Causes Eventus to store the two-digit share code from the CRSP name history array as the variable SHRCODE in any OUTSAS data set.

SIC Causes Eventus to report the Standard Industrial Classification code from the CRSP name history array and to add the variable SICCODE to any output data set. The code is reported on the estimation period statistics listing.

SKIP=*n* Gives the number of header lines in the USERSTOK external data file. Eventus ignores the header lines.

SPORT When the EXCESS option appears on the EVENTUS statement, this option switches from beta-based to standard deviation-based excess returns or decile index portfolios.

STDALL Requests that the standardized test statistic be computed for the non-market model benchmark(s) as well as for the market model. The STDALL option is ignored in TWIN event studies. Any applicable standardized-test options then affect all benchmarks in use. For example, the SERIAL, STDCSECT, and BOOT options affect standardized tests for market adjusted returns as well as market model returns when STDALL is in effect.

STDCSECT Specifies that the standardized cross-sectional test (Boehmer, Musumeci and Poulsen, 1991) be substituted for the Patell z test in the standardized method. This option implies SERIAL option.

STDONLY Please see NOSTD.

SUR Causes Eventus to use the seemingly unrelated regressions approach, where the market model, augmented by a dummy variable for each window specified on the WINDOWS statement, is estimated by joint generalized least squares over the combined estimation period and event period return series.

SW Produces Scholes-Williams (1977) market model results in addition to the OLS results. The EGARCH, GARCH and SW options are mutually exclusive.

TAIL=1|2 Specifies the significance levels of the reported test statistics are to be based on one or two tailed tests. The default is TAIL=1. The

TAIL option does not set the tails for bootstrapped tests; please see the BTAIL option.

TIMEUNIT=*n* Specifies that Eventus should aggregate returns in the estimation and event periods into *n*-day (when the original returns are daily) returns before performing the analysis. When n is even, period 0 in event time contains day 0 (the date in the request file), $\frac{n}{2} - 1$ days following day 0 and $\frac{n}{2}$ preceding day 0. Additional periods are formed on either side of period zero.

Eventus combines daily, weekly, monthly or quarterly returns into period returns by addition. If the LOG option is specified on the REQUEST statement, individual daily or monthly returns are converted to logarithmic form before adding.

Eventus interprets the WINDOWS statement arguments and the PRE, POST and MAXMISS options or their defaults in terms of multiday, multiweek or multimonth periods. All results are reported in terms of multiday periods as well. However, Eventus interprets the EST, ESTLEN, MINESTN and other REQUEST statement options in terms of the original days, weeks, months, or quarters.

VALUE|BOTH By default, Eventus uses only equally weighted market index returns in the market model and market adjusted returns event studies. Specify VALUE to change to the value weighted index (or the alternative index indicated by SP500 or COMPOSIT specified on the REQUEST statement) or BOTH to produce separate event studies using both indexes.

B.5 EXTRACT Statement

Extracts window cumulative or compounded abnormal returns and optional weighted least squares regression weights from a SAS data set previously saved by the Eventus EVTSTUDY statement with the OUTSAS option. The following parameter is mandatory.

INSAS=*libref.membername* Points to the SAS data set containing the saved event study results.

The following options can be specified on the EXTRACT statement.

BUYHOLD Specifies buy-and-hold compounded return computation for windows instead of the default additive cumulation.

CDCSI Causes Eventus to compute regression weights to reflect the weighting implied by the Collins and Dent (1984) test assuming cross-sectional independence. This option affects only the weight, so it is not applicable unless the WPREFIX option also is specified.

EXTEND=*n* If there are missing returns in a window for a security, Eventus attempts to "fill out" the window by searching *n* trading days (or weeks, months, quarters or years) following each window for nonmissing returns. The default is EXTEND=0, indicating no filling out.

EXTFILE=*fileref|file name* Gives the fileref or name of the external file in which Eventus is to store the converted request file. If the argument is a file name, it may include the path but must not include any blank or period. On most systems, a .dat extension will be added to the file name automatically. The default is USERDATA. On mainframe systems where the file is pre-allocated by control language or in the filename statement, the file should be given a disposition of NEW or OLD, a fixed block format, and a logical record length of 80.

ID=*variable* Names the variable to be used as an observation (event) identifier. The variable must exist on the INSAS data set. Specify IDFMT (see below) also.

IDFMT=*format* Gives the output format of the identifying variable named in theID option. Any SAS format is acceptable.

OUTSAS=*libref.membername* Indicates that the output should be in the form of a SAS data set and names the data set to be created within the aggregate storage location associated with *libref*. The default is to create an external file (ASCII or EBCDIC text or binary) file.

SERIAL Selects WLS regression weights based on the serial dependence-corrected standard error in equation A.1 on page 108. The default is to produce weights that provide regression weighting consistent with

the portfolio weighting implicit in the non-dependence-corrected multiperiod Patell (1976) test. No weights are actually created unless the WPREFIX option also is specified.

TYPE=CP|MAR|MM|RAW|SW Selects comparison period mean adjusted, market adjusted, market model adjusted, raw, or Scholes-Williams market model adjusted returns for output. The selected type must exist on the input SAS data set. The default is TYPE=MM.

VALUE Selects value-weighted-index (or S & P 500 or Nasdaq Composite Index) based abnormal returns when the EVTSTUDY statement that created the input SAS specified the BOTH option. The default is to use equally-weighted-index based abnormal returns in this situation.

VPREFIX=*prefix* Sets the variable name prefix to use for the window abnormal returns. Acceptable values of *prefix* are valid SAS names up to six characters long. Longer values will be truncated to six characters. The default prefix is WINAR.

WPREFIX=*prefix* Causes Eventus to include WLS regression weights in the output and sets the variable name prefix to use for the weights. Acceptable values of *prefix* are valid SAS names up to six characters long, or up to seven characters if fewer than ten windows are to be output. The default is not to output weights; there is no default prefix.

B.6 OLDSTUDY Statement

The OLDSTUDY statement reprints the reports from previous runs of Eventus and creates new reports by merging two or three OUTSAS= data sets from previous runs into a single sample.

ALLDAYS If the number of days covered by the event study, PRE+POST+1, exceeds 98, not all the individual day portfolio returns are listed by default. Day 0 is always listed, together with a proportionate number of contiguous PRE and POST days. All the windows you request are printed, regardless of their date ranges. To obtain a complete printout of all the individual days' portfolio returns, specify ALLDAYS. This option is ignored for TWIN event studies.

BOOT Requests bootstrap tests for window abnormal returns. Bootstrap tests are in addition to the normally reported parametric and nonparametric tests. `BOOT` implies the `BUYHOLD` and `STDCSECT` options. It is not possible in the current version of Eventus to bootstrap `TWIN` event studies. The option is not available unless it was specified on the original `EVTSTUDY` statement.

BTAIL=1|2 Selects one- or two-tailed bootstrap tests. The default is 2. This option has no effect unless the `BOOT` option also is specified.

BUYHOLD Specifies buy-and-hold compounded return computation for windows instead of the default additive cumulation. The option is not available unless it was specified on the original `EVTSTUDY` statement.

CDCSI Requests the Collins and Dent (1984) test assuming cross-sectional independence instead of the default standardized test. The option is not available unless it or the `EGLS` option appeared on the original `EVTSTUDY` statement.

CSECTERR Substitutes a cross-sectional standard deviation for the default time-series standard deviation in non-standardized t-statistic computations.

DETAIL|DETAIL=FULL See `EVTSTUDY` above. These need not have been specified on the original `EVTSTUDY` statement in order to work here.

ID=*variable* Names the variable to be used as an observation (event) identifier. The variable must exist on the `INSAS` data set.

INSAS[2|3]=*libref*[*.membername*] Selects the Eventus abnormal return data set to use. `INSAS2=` and `INSAS3=` optionally select additional data sets to combine with the first data set to run a merged event study. Each of these must first be created by the Eventus `EVTSTUDY` statement using `OUTSAS=` and `PACKAGE=1`. All data sets to be combined must have had the same `ESTLEN` (if any) on the original `REQUEST` statement.

JACKNIFE Designates the jackknife test instead of the generalized sign test as the nonparametric test to accompany non-standardized method parametric tests. The option is valid only if it also appeared on the original `EVTSTUDY` statement.

MEDIAN This option affects printed output, but only when the PAGE=WIDE option is specified on the EVENTUS statement. The MEDIAN option causes the daily (or monthly) median abnormal return to be printed instead of the number positive and negative and nonparametric test. The option affects only the daily or monthly output, not the output for windows.

NOPLIST Suppresses the printing of the estimation period statistics that normally appear between the input report and the event study results.

NOSTD|STDONLY NOSTD omits standardized abnormal returns from the output, provided that STDONLY was not specified with the original event study. STDONLY prints only market model standardized abnormal return tests, provided market model abnormal returns were included in the original event study.

RANKTEST Designates the rank test, instead of the generalized sign test, as the nonparametric statistic to appear with the non-standardized parametric tests. The option is valid even if it was not specified on the original EVTSTUDY statement.

STDALL Specifies that the standardized test statistic be computed for the non-market model benchmark(s) as well as for the market model. The option is ignored in TWIN event studies. All applicable standardized test options then affect all benchmarks in use. For example, the SERIAL, STDCSECT, and BOOT options affect standardized tests for market adjusted returns as well as market model returns when STDALL is in effect.

STDCSECT Specifies that the standardized cross-sectional test (Boehmer, Musumeci and Poulsen, 1991) be substituted for the Patell z test in the standardized method. This option is valid only if it was specified on the original EVTSTUDY statement. This option implies SERIAL option.

SW Produces Scholes-Williams (1977) market model results in addition to the OLS results, but only if the option also appeared on the original EVTSTUDY statement.

TAIL=1|2 Specifies the significance levels of the reported test statistics are to be based on one or two tailed tests. The default is TAIL=1. The TAIL option does not set the tails for bootstrapped tests; please see the BTAIL option.

B.7 PRICES Statement

Reads prices from CRSP daily or monthly stock databases. Eventus writes the prices and associate data to the SAS output window or stores them in a SAS data set or external file. Choose from the following options.

BIDASK Specifies that the secondary price variables are to be read and included in the output. Secondary prices are either bid and ask quotes or intraday/intramonth high and low transaction prices; see the CRSP *Data Description Guide* for details.

BINARY This is identical to the TEXT format below, except that the prices themselves are written in real binary (FLOAT4., RB4., or S370RB4. depending on the system) format. Real binary format uses about one third as much disk space for storing prices as the TEXT format. A binary file, like a TEXT file, can be read by any computer language, including FORTRAN, C, and SAS.

DISTRIB Requests that dividends and other cash distributions (CRSP distribution code ≤ 4999) be extracted. When the DISTRIB option is in effect, the printed or SAS data set output includes one zero or nonzero value for each trading date, containing the total cash distribution per share for which the stock went ex on that date. Two or more cash distributions with the same date are added and the total reported. If the SPLITADJ option also is specified, the reported distribution total reflects the same split adjustment as the price. The DISTRIB option is not available when the HSAS option is specified unless only one trading day (or month) per stock is being extracted. The variables DISTCODE, containing the four-digit CRSP distribution code, and DIVAMT, containing the cash distribution per share, are added to any output data set.

EXTFILE= Gives the fileref of the external file in which Eventus is to store the prices. This option is only valid with the default TEXT or optional BINARY file format options. If EXTFILE is not specified, the prices will be printed to the fileref USERDATA.

HSAS Creates a horizontal format SAS data set. A SAS data set cannot be read by non-SAS software. The data set will contain as many of the following variables as applicable: PRI1 through PRI*nnnn*, BIDL1 through BIDL*nnnn*, ASKH1 through ASKH*nnnn*, TRAD1 through TRAD*nnnn*, PERMNO,

EVENTDA1, EVENTDA2, SHARES and any ID= variable specified on the REQUEST statement. You must specify the SAS data set name using the OUTSAS= option.

EVENTDA1 and EVENTDA2 are SAS date variables representing the beginning and ending trading days of the interval you request. The one to three digit number *nnnn* is the maximum number of trading days in any of the intervals you requested. For example, if the number of trading days you ask for ranges from 2 for some firms to 90 for others, *nnnn* is 90. For a firm in this example with only two trading days, PRI3 through PRI90 contain missing values. Because the SAS system stores a large amount of information on each variable for its own use, a horizontal format data set occupies considerably more disk storage space than a vertical format data set (see the VSAS option.)

NMS Instructs Eventus to obtain bid and ask quotes (if you also specify BIDASK and the number of trades (if you also specify TRADES) from the Supplemental Nasdaq Data Arrays. When available, quotes from the Supplemental Nasdaq Data Arrays (CRSP names *Bid* and *Ask*) replace secondary prices (CRSP names *Bid or Low Price* and *Ask or High Price*.) To prevent the replacement of secondary prices, and instead have Eventus report both secondary prices and Supplemental Nasdaq quotes when the latter are available, specify NMS NOCLOSE on the PRICES statement.

When both NMS and NOCLOSE appear on the PRICES statement, Eventus produces an output file or data set that contains both the secondary prices and the Supplemental Nasdaq quotes. The secondary prices are identified as BID OR INTRADAY LOW and ASK OR INTRADAY HIGH in an external file, or by the variable names BIDLO and ASKHI in a vertical SAS data set or BIDL1 through BIDL*nnnn* and ASKH1 through ASKH*nnnn* in a horizontal SAS data set. The Supplemental Nasdaq quotes are identified as BID FROM NMS FILE and ASK FROM NMS FILE in an external file or by variable names BIDNM and ASKNM or BIDN1 through BIDN*nnnn* and ASKN1 through ASKN*nnnn* in a vertical or horizontal SAS data set, respectively.

When the option combination BIDASK NMS is in effect without NOCLOSE, Eventus reports only one pair of merged secondary price variables, containing Supplemental Nasdaq bid and ask quotes for those stocks and

date ranges for which the supplemental quotes are available, and the *Bid or Low Price* and *Ask or High Price* data items from the main CRSP secondary prices arrays in all other cases. The merged secondary price variables are identified as `BID OR INTRADAY LOW` and `ASK OR INTRADAY HIGH` or by variable names `BIDLO` and `ASKHI` or `BIDL1` through `BIDL`*nnnn* and `ASKH1` through `ASKH`*nnnn*. The CRSP *Data Description Guide* for the stock database provides a full explanation of the underlying stock data.

NOCLOSE Specifies that primary closing prices are to be omitted; used together with the `BIDASK` option when only secondary prices are desired. Note the special interaction, explained above, of this option and the `NMS` option.

OUTSAS= Specifies a two-level SAS name (*libref.membername*) under which to create the SAS data set containing the prices. This option is valid only with the `HSAS` or `VSAS` file format options.

SHARES Specifies that the number of shares outstanding on the event date, in thousands, should be included in the output file. This option is ignored unless either `NDAYS=1` is specified on the `REQUEST` statement, or `VSAS` is specified on the `PRICES` statement. The shares outstanding data may not be as timely as the price data; refer to CRSP documentation for more information.

SHRCODE Causes Eventus to store the two-digit share code from the CRSP name history array as the variable `SHRCODE` in any `OUTSAS` data set.

SIC Causes Eventus to read the SIC code from the CRSP database and to add the variable `SICCODE` to any output SAS data set.

SPLITADJ Causes Eventus to adjust for stock splits, reverse splits, and stock dividends using distribution data from the CRSP database. The adjustment takes place only within the range of data being extracted for each stock. For splits and stock dividends occurring after the first date extracted, Eventus multiplies prices and cash distributions by a split factor, and divides shares outstanding by the same factor. The split factor is equal to 1.0 on the first date being extracted and is cumulative within the range of data being extracted. The split factor increases by a factor of 1+ the CRSP "factor to adjust price" each time there is a

split or stock dividend. The split factor (variable SPLFAC) is added to any output SAS data set. The SPLITADJ option is not supported with the HSAS output option.

TEXT This is the default. Eventus writes the prices on a file that people (not just computers) can read. The data are arranged in columns, with each row reporting the PERMNO, identification variable if applicable, CRSP trading day or month number, and prices. All the data for the first stock are listed, with each trading day on its own row, followed by all the data for the second stock, and so on.

Specify the fileref (SAS file shortcut, or a DDname on a mainframe system) to which to write the prices with the EXTFILE= option on the PRICES statement. On mainframe systems that require files to be pre-allocated, the file pointed to by the fileref should have a logical record length of 80 or longer, a fixed block format, and a block size that conforms to your system's rules.

TRADES This option is ignored unless NMS is specified on the PRICES statement. Specifies that the number of trades on each date should be read from the Supplemental Nasdaq Data and included in the output file.

VSAS Creates a vertical format SAS data set. A SAS data set cannot be read by non-SAS software. The following variables are included in the data set: PERMNO, CUSIP if the CUSIPERM option is in effect, CRSPDAY1, DATE, BIDLO and ASKHI if applicable, PRICE, SHARES when applicable, TRADES when applicable, and any identifying variable you list on the REQUEST statement. CRSPDAY1 is the CRSP day (or month, *etc.*) number for the beginning date you request; DATE is the actual date of each price, recorded as a SAS date variable. PRICE is the security price, and SHARES is the number of shares outstanding in thousands.

There is one observation in the SAS data set for each trading day in the interval you request. Use the VSAS option if you plan to process the data with a SAS program that uses BY variable techniques. Specify the SAS data set name using the OUTSAS= option.

B.8 REQUEST Statement

This statement is required after the EVENTUS statement and before the EVTSTUDY, RETURNS, PRICES or VOLUME statement. It is only used if the EVTSTUDY, RETURNS, PRICES or VOLUME statement is used. It reads the user request file and the CRSP index file and merges the two. The user may specify these options:

AUTODATE Specifies that a calendar date in the request file that is not a trading day should be converted to the following trading day. Eventus converts a date in the request file that follows the last trading day listed in the crspindx file to a SAS missing value, which may cause unexpected error messages.

AUTODATE=BACK Specifies that a calendar date in the request file that is not a trading day is to be converted to the preceding trading day. Eventus converts a date in the request file that precedes the first trading day listed in the crspindx file to a SAS missing value, which may cause unexpected error messages.

COMPOSIT See SP500.

CUSIP Specifies that the CRSP database being used is sorted (either sorted or indexed if in SAS data set form) by CUSIP and the request file contains CUSIPs instead of PERMNOs. The default is that the CRSP database is sorted by PERMNO and the request file contains PERMNOs.

CUSIPERM Specifies that the request file contains CUSIPs but does not change the default assumption that the CRSP database is sorted by PERMNO. Eventus will attempt to convert CUSIPs to PERMNOs during execution. This requires conversion files that normally are created during installation or upgrading. If these files have not been created, please see the Eventus installation instructions for the required procedure.

DATEFMT=*format* Specifies the format of the dates in the request file. The specification must be either a valid SAS date informat or the word CRSP. The word CRSP tells Eventus to look for a 1–to–4 digit integer representing a CRSP trading day (or month) number (1=July 2, 1962 for all the daily files, except 1=December 14, 1972 for the old format (1985 and 1987) of the Nasdaq files.) Leading zeroes need not (but

may) be included in the CRSP day number. Any format other than CRSP must be a valid SAS date format (although the period at the end is optional.) The default is `DATEFMT=YYMMDD8`.

EST=*periods* and POOL The `EST` option lets you choose the estimation period Eventus uses to estimate the benchmark return parameters for the event study. If you specify a negative value, Eventus constructs an estimation period ending that number of trading days (months) *before* the event date. If you give a positive number (with or without the plus sign), the estimation period follows the event date and begins that number of trading days (months) *after* day (month) zero. For example, `EST=+61` means that the estimation period will begin 61 trading days after the event date, while `EST=−80` tells Eventus to end the estimation period on event-time date −80. For a `TWIN` event study, Eventus considers negative values relative to the first event date, positive values relative to the second.

- If you specify `POOL`, Eventus constructs the estimation period using equal numbers of returns from two periods: one ending 61 days before the (first) event date, and one beginning 61 days after the (second) event date. You can control the length of the combined period with the `ESTLEN` parameter (described below.) For example, `ESTLEN=200` with `POOL` gives you an estimation period consisting of days −160 through −61 and +61 through +160. Do not specify `POOL` with `EST=SPECIFIC`.
- If you specify `EST=SPECIFIC`, Eventus looks for a second date in the request file (or a third date in a `TWIN` event study), and *ends* the estimation period on the date specified for each firm. This allows you to choose estimation periods that fall a varying number of days from the event date, according to the circumstances of each case. When using `EST=SPECIFIC` with external request files, make sure that the estimation date is the next item after the event date on each line. If the request file is a SAS data set (specified by `INSAS=`), the specific estimation date must be a variable named `ESTEND` (`CRSPEST` if `DATEFMT=CRSP` is in effect.) Do not specify `POOL` with `EST= SPECIFIC`.
- The default is `EST=−46` for daily event studies, `EST=-10` for weekly event studies and `EST=−13` for monthly event studies.

ESTLEN=*n* Specifies the length of the estimation period in trading days, weeks, months, quarters or years, depending on the return interval being used for estimation in the current run. The maximum permitted is 999. The default is 255 days (about one year) for daily returns, 60 months for monthly returns, 52 weeks for weekly returns, 20 quarters for quarterly returns, or 10 years for annual returns. Odd values of `ESTLEN` will be reduced by one day when `POOL` is specified. The number of usable returns in the estimation period may be lower than `ESTLEN` in individual cases if there are missing returns on the CRSP database.

GROUP=*group_variable* Names a grouping variable to be used in an event study to combine multiple observations into a single equally weighted portfolio. The value of the grouping variable for each observation is listed on the appropriate line of the request file after the dates and `ID` variable (if any.) The grouping variable must be an integer between 0 and 9999 inclusive; leading zeros in the request file are optional and ignored. Two or more observations with identical grouping variable values are combined into a portfolio treated as a single observation. The default is to use no grouping variable so that each observation is weighted equally.

GRWEIGHT Valid only if the `GROUP` option is specified. Denotes that the request file contains a group weight variable. This variable, expressed as a decimal, specifies the weight to be given the individual stock within its group portfolio. All the weights for a single group should sum to 1. If the request files is a SAS data set, the variable name of the group weight must be `grweight`.

ID=*variable* Names the variable to be used as an observation (event) identifier. The identifying variable may be of any data type. If you also specify `INSAS`, the identifying variable must exist on the permanent SAS data set specified. Specify `IDFMT` (see below) also.

IDFMT=*format* Gives the format of the identifying variable in the external request file. For example, if your identifying variable is a one to four digit integer, specify `IDFMT=4.`.

INSAS=*libref.membername* Used when the request file is a SAS data set. The data set must contain these variables:

- PERMNO (numeric variable containing the 5-digit CRSP permanent number), unless the CUSIPERM option or CUSIP option is specified, or the NONCRSP option is in effect.
- CUSIP (character variable of length eight containing the CUSIP identifier for CRSP input or containing any identifier for non-CRSP input.) The CUSIP variable on the input data set is ignored unless the CUSIPERM option or CUSIP option is specified, or the NONCRSP option is in effect.
- For single event date event studies only, either EVENTDAT (SAS date variable containing day 0) or CRSPDAY (integer representing trading date 0 as a sequence number corresponding to the CRSP calendar. The integer may represent the day, week, month, quarter or year from the beginning date of the index file.)
- For TWIN event studies, and for data retrieval using RETURNS or PRICES, the two SAS date variables EVENTDA1 and EVENTDA2, must be present, or if DATEFMT=CRSP is in effect, then the variables CRSPDAY1 and CRSPDAY2 must be present. However, if the NDAYS= option is specified, then omit EVENTDA2 or CRSPDAY2.
- If you use EST=SPECIFIC (see below), either ESTEND (SAS date variable) or CRSPEST (CRSP day, week or month number) must also be present on the data set.
- If you specify ID=*variable* (see below), the *variable* you name.
- If GROUP=*group_variable* is specified, then the name given in place of *group_variable* must be present.
- If GRWEIGHT is specified, then a numeric variable named GRWEIGHT, containing the weight of the firm-event within the group, must be present.
- If the NAME option appears, a character variable of length 33, NAME, must be present.
- If the SHORT option is specified, a character variable of length one, named SL, must have a value of either 'S' or 'L'.
- If DATEFMT=CRSP (see above) is in effect, the event day or month number (CRSPDAY), or, for a TWIN event study, the beginning and ending day or month numbers (CRSPDAY1 and CRSPDAY2.) If

EST=SPECIFIC is also in effect, the ending day or month number of the estimation period must be included as variable CRSPEST.

INSAS2=*libref.membername* Applicable only when you specify DUAL on the EVENTUS statement; used in place of REQFILE2 when the request files are SAS data sets. The format of the second request file must be exactly the same as the format of the first request file; see the INSAS option above.

IX2Y Specifies that the SFA or Eventus format size index file being read in the current run uses two digit years. The default is to expect four digit years.

LOG Causes daily or monthly returns to be transformed to continuously compounded returns by taking $\ln(R_{jt})$ for stock returns and $\ln(R_{mt})$ for market index returns.

MINESTN=*n* Optionally specifies the minimum number of usable trading days in the estimation period. (Eventus considers an estimation period return usable if it is non-missing, except for the first return following a sequence of one or more missing returns.)

NAME Indicates that a firm name appears as the last item on each line of the request file. The option is ignored everywhere but non-CRSP event studies.

NDAYS=*n* An option on REQUEST statements in conjunction with RETURNS, PRICES and VOLUME statements, and TWIN event studies. Specifies that you want to retrieve *n* consecutive trading days of data or months (including days or months with missing data values) for each observation. If you specify NDAYS, the first day or month is the date in the request file; omit the ending date, which must be specified in the above situations when NDAYS is not used, from the request file.

NODIVIDX Causes Eventus to market indices excluding dividends instead of the default indices including dividends.

POOL See EST.

PORTYPE=1|2|... Specifies the portfolio-type position within a CRSPAccess portfolio assignment structure to which Eventus reads when extracting decile ranks for the SIZEINDX option. Users should not specify this option except as explicitly directed by Eventus documentation.

REQFILE=*fileref* The most often used method of storing the request information (PERMNO identifiers, dates, etc.) is to use an external (non-SAS formatted) file, typically an ASCII text file. Replace *fileref* with the fileref associated with your file. The REQFILE specification may be omitted if the fileref of the request file is REQUEST. Each line of the request file should contain the following variables in order:

```
PERMNO event  [event       [specific     [ID number  [grouping  [group  [S or
        date   date 2]  estimation date]  or string]  variable]  weight]  L]
```

Each value must be separated by at least one blank, but the exact position of the values is unimportant as long as they appear in the order shown. The square brackets simply indicate items that need not always appear; do not include them in the file. Whether an optional item should appear is determined by the options specified on the EVENTUS and REQUEST statements. The file need not be sorted by PERMNO. If the CUSIPERM option or the CUSIP option is specified, the first variable on the line should be CUSIP instead of PERMNO.

REQFILE2=*fileref* Used to specify the fileref of the second request file when DUAL is specified on the EVENTUS statement; the default is REQFILE2=REQUEST2.

SHIFT1=n_1,SHIFT2=n_2 The SHIFTn options are intended primarily for the DATECONV statement and for the REQUEST statement in a RETURNS or PRICES program, but may also be used on the REQUEST statement in a TWIN EVTSTUDY program. The first date in the request file is shifted by n_1 periods and the second date is shifted by n_2 periods. For the monthly file, the periods are months. For the daily file, the periods are trading days if DATEFMT=CRSP and calendar days otherwise. Both n_1 and n_2 may be specified as any integer value. For example, SHIFT1=-1 shifts June 1, 2000 back to May 31, 2000.

SHIFT1 and SHIFT2 may be specified singly as well as together. Using these options with calendar dates may result in invalid date messages

unless AUTODATE is also specified. If the researcher needs to shift by trading days rather than calendar days, it may be necessary to convert to CRSP trading day numbers using DATECONV first.

SHORT Specifies that an S (for short position) or L (for long position) code appears at the end of each line in the request file. When S appears, all raw stock returns and index returns for that event are multiplied by −1 before any analysis.

SIZEINDX[=CRSP]Causes Eventus to use a size index file for the market index return in event studies. The size decile is matched to the size portfolio number in the annual data structure of the CRSP stock file. If SIZEINDX=CRSP is specified, Eventus expects the fileref sizeindx to be associated with the character (not binary) CRSP Indices/Decile file. If only SIZEINDX is specified, Eventus expects the fileref sizeindx to be associated with a binary or character file containing the date followed by the returns on size decile portfolios 1–10. Such a file is generated by a program included with the Eventus installation kit. If the main CRSP database and size index file are in the form of SAS data sets, please see the EVENTUS statement option SIZEIND.

SP500|COMPOSIT SP500 specifies to read the return on the Standard and Poor's 500 Composite Index instead of the value weighted index of all stocks, regardless of whether NODIVIDX is specified.COMPOSIT tells Eventus to read the return on the Nasdaq Composite Index instead of the value weighted index of all stocks, regardless of whether NODIVIDX is specified. With SFA CRSP files, if the index file associated with the fileref crspindx is a NYSE-AMEX or NYSE-AMEX-Nasdaq index file, the Standard and Poor's Composite Index is read instead, but Eventus output will incorrectly identify it as the Nasdaq Index.

UPCUSIP This option, valid only when CUSIP also is specified, makes Eventus attempt to update CUSIP identifiers in the request file to match the latest version of the CRSP stock file before searching CRSP. Requires a conversion file, a component of Eventus that normally should have been created when the software was installed or last upgraded.

B.9 RETURNS Statement

Reads returns from a CRSP stock database. No processing occurs, except that missing returns are converted from CRSP to SAS missing value codes. Eventus stores the returns in your choice of file. Choose from the following options.

BINARY This is identical to the TEXT format, except that the returns themselves are written in real binary (FLOAT4., RB4., or S370RB4. depending on the system) format. This uses only one third as much disk space for storing returns as the TEXT format. This type of file, like the TEXT file, can be read by any major computer language, including FORTRAN, SAS and C.

BOTH See VALUE.

EXTFILE= Gives the fileref of the external file in which Eventus is to store the returns. The default is userdata. On mainframe systems where applicable, the file should have a disposition of NEW or OLD and a logical record length of 80. This option is only valid with the default TEXT or optional BINARY format.

HSAS Creates a "horizontal format" SAS data set. A SAS data set cannot be read by non-SAS software. The data set will contain as many of the following variables as applicable: RETN1 through RETN*nnn*, MKT1 through MKT*nnn* if you specify INDEX on the RETURNS statement, PERMNO, EVENTDA1, EVENTDA2, and any identifying variable that you list on the REQUEST statement. If you specify the BOTH option, value weighted index returns will be stored as VWMK1 through VWMK*nnn*. You must specify the SAS data set name using the OUTSAS= option.

EVENTDA1 and EVENTDA2 are SAS date variables representing the beginning and ending trading days of the interval you requested. The one to three digit number *nnn* is the maximum number of trading days in any of the intervals you requested. For example, if the number of trading days you ask for ranges from 2 for some firms to 90 for others, *nnn* is 90. For a firm in this example with only two trading days, RETN3 through RETN90 contain missing values. Because the SAS system stores a large amount of information on each variable for its own use, a horizontal

format data set occupies considerably more disk storage space than a vertical format data set.

INDEX Includes market index returns in the file. The default is not to include market index returns.

OUTSAS= Specifies a two-level SAS name (*libref.membername*) under which to create the SAS data set containing the returns. This is valid only with the `HSAS` or `VSAS` option.

SHRCODE Causes Eventus to store the two-digit share code from the CRSP database's name history array as the variable `SHRCODE` in any `OUTSAS` data set.

SIC Causes Eventus to read the SIC code from the CRSP database and to add the variable `SICCODE` to any output SAS data set.

TEXT This is the default. Eventus writes the prices on a file that people can read. The data are arranged in columns, with each row reporting the PERMNO, identification variable if applicable, crsp date, and return. All the data for the first stock are listed, with each trading day or month on its own row, followed by all the data for the second stock, and so on.

Specify the file name or fileref (a SAS file shortcut, or on mainframe systems, a DDname) to which to write the returns with the `EXTFILE=` option on the `RETURNS` statement. If the argument of `EXTFILE` is a file name, it may include the path but must not include any blank or period. On most systems, a `.dat` extension will be added to the file name automatically. On mainframe systems where files must be pre-allocated, the file should have a logical record length of 80 or longer, a fixed block format, and a block size that conforms to your system's rules.

VALUE|BOTH In the absence of one of these options, Eventus includes only equally weighted market index returns in the output when the `INDEX` option is specified. Specify `VALUE` to change to the value weighted index (or the alternative index indicated by `SP500` or `COMPOSIT` specified on the `REQUEST` statement) or `BOTH` to get both.

VSAS Creates a vertical format SAS data set. A SAS data set cannot be read by non-SAS software. The following variables are included in the data set: `PERMNO`, `CUSIP` if the `CUSIPERM` option is in effect, `CRSPDAY1`, `DATE`, `RETURN`, and any identifying variable you list on the `REQUEST` statement. If you specify `INDEX` on the `RETURNS` statement, `MARKET` and `_WEIGHT_` are also included. `CRSPDAY1` is the CRSP trading day number for the beginning date you requested; `DATE` is the actual date of the return, recorded as a SAS date variable. `RETURN` is the security return, and `MARKET` is index return (including dividends.) The variable `_WEIGHT_` is a character variable of length 5, equal to `'Equal'`, `'Value'`, `'NASDQ'` or `'SP500'`.

There is one observation in the data set for each trading day in the interval you request. Use the `VSAS` option if you plan to process the data with a SAS program that uses BY-variable techniques. Specify the SAS data set name using the `OUTSAS=` option.

B.10 `TITLE` and `TITLE2` Statements

`TITLE` statements are optional; they specify page headings, mainly for event study output. Enter title statements after the `EVENTUS` statement. `TITLE` and `TITLE2` are regular SAS statements, not part of Eventus as such. The default title is EVENTUS. Eventus does not use additional title statements (`TITLE3`, etc.); if you specify them they are ignored.

B.11 `UPCUSIP` Statement

This statement is used after the `EVENTUS` statement to update CUSIP identifiers to match the latest version of the CRSP stock file installed at your site. Now that CRSP stock files are sorted by PERMNO, the `UPCUSIP` statement is becoming obsolete. It still is available for users who may be working with 1993 or earlier releases of the CRSP stock files. The statement reads the user request file and performs the updating operation. The user may specify these options:

COLUMN=*n* Use when the CUSIP is not the first item on each line of the request file. Substitute the starting column for the CUSIP for *n*. This option is not needed if only blanks precede the CUSIP on each line.

EXTFILE= Gives the fileref of the external file in which Eventus is to store the updated copy of the request file. The default is `userdata`. On mainframe systems where applicable, the file should have a disposition of `NEW` or `OLD` and a logical record length of 80.

REQFILE=*fileref* The request file for `UPCUSIP` must be an external (non-SAS formatted) file, such as a card image format file. Replace *fileref* with the fileref associated with your file. The `REQFILE` specification may be omitted if the fileref of the request file is `REQUEST`. The file need not be sorted by CUSIP.

B.12 `VOLUME` Statement

Reads trading volume from the CRSP stock files. Eventus stores the volume data in your choice of disk file. Choose from the following options.

BINARY This is identical to the `TEXT` format, except that the volume data are written in real binary (`FLOAT4.`, `RB4.`, or `S370RB4.` depending on the system) format. This uses only one third as much disk space for storing volume data as the `TEXT` format. This type of file, like the `TEXT` file, can be read by any major computer language, including FORTRAN, SAS and C.

EXTFILE= Gives the name or fileref of the external file in which Eventus is to store the volume data. If the argument of `EXTFILE` is a file name, it may include the path but must not include any blank or period. On most systems, a `.dat` extension will be added to the file name automatically. On mainframe systems where applicable, the file should have a disposition of `NEW` or `OLD`, a fixed block format, and a logical record length of 80. The `EXTFILE` option is not for use with the `VSAS` and `HSAS` options, which select SAS data set rather than external file output.

HSAS Creates a "horizontal format" SAS data set. A SAS data set cannot be read by non-SAS software. The data set will contain as many of the following variables as applicable: `VOL1` through `VOL`*nnn*, `PERMNO`, `EVENTDA1`, `EVENTDA2`, `SHARES` and any identifying variable you list on the `REQUEST` statement. You must specify the SAS data set name using the `OUTSAS=` option.

EVENTDA1 and EVENTDA2 are SAS date variables representing the beginning and ending trading days of the interval you request. The one to three digit number *nnn* is the maximum number of trading days in any of the intervals you requested. For example, if the number of trading days you ask for ranges from 2 for some firms to 90 for others, *nnn* is 90. For a firm in this example with only two trading days, VOL3 through VOL90 contain missing values. Because the SAS system stores a large amount of information on each variable for its own use, a horizontal format data set occupies considerably more disk storage space than a vertical format data set.

NASDINFO Retrieves trading status trait code, NMS indicator and market maker count from the Nasdaq information arrays of the CRSP database. Valid only when SHARES (see below) is a valid option.

OUTSAS= Gives the two-level SAS name (*libref.membername*) in which to store the volume data. A SAS libname statement or the operating system must associate the libref with the aggregate storage location used for the SAS data library. The data set is created if it does not exist already, or replaced if it does exist.

SHARES Specifies that the number of shares outstanding on the event date, in thousands, should be included in the output file. This option is ignored unless either NDAYS=1 is specified on the REQUEST statement or VSAS is specified on the VOLUME statement. Note that the shares outstanding data may not be as timely as the volume data. Refer to CRSP publications for more information.

SHRCODE Causes Eventus to store the two-digit share type code from the CRSP name history array as the variable SHRCODE in any OUTSAS data set.

SIC Causes Eventus to read the SIC code from the CRSP database and to add the variable SICCODE to any output SAS data set.

SPLITADJ Causes Eventus to adjust for stock splits, reverse splits, and stock dividends using distribution data from the CRSP database. The adjustment takes place only within the range of data being extracted for each stock. For splits and stock dividends occurring after the first date extracted, Eventus multiplies share volume and cash distributions by a

split factor, and divides shares outstanding by the same factor. The split factor is equal to 1.0 on the first date being extracted and is cumulative within the range of data being extracted. The split factor increases by a factor of 1+ the CRSP "factor to adjust price" each time there is a split or stock dividend. The split factor (variable `SPLFAC`) is added to any output SAS data set. The `SPLITADJ` option is not supported with the `HSAS` output option.

TEXT This is the default. Eventus writes the prices on a file that people can read. The data are arranged in columns, with each row reporting the PERMNO, identification variable if applicable, crsp date, and volume. All the data for the first stock are listed, with each trading day on its own row, followed by all the data for the second stock, and so on.

Specify the fileref (a SAS file shortcut or, on mainframe systems, a DDname) to which to write the prices with the `EXTFILE=` option on the `VOLUME` statement. On a mainframe operating system that defines record lengths and blocks, the file should have a logical record length of 80 or longer, a fixed block format, and a block size that conforms to your system's rules.

TRADES Specifies that the number of trades on each date should be read from the Supplemental Nasdaq Data Arrays and included in the output file.

VSAS Creates a "vertical format" SAS data set. A SAS data set cannot be read by non-SAS software. The following variables are included in the data set: `PERMNO`, `CUSIP` if the `CUSIPERM` option is in effect, `CRSPDAY1`, `DATE`, `VOLUM`, `SHARES` when applicable, and any identifying variable you list on the `REQUEST` statement. `CRSPDAY1` is the CRSP "day number" for the beginning date you request; `DATE` is the actual date of the volume figure, recorded as a SAS date variable. `VOLUM` is the trading volume in shares, and `SHARES` is the number of shares outstanding in thousands.

There is one observation in the SAS data set for each trading day in the interval you request. Use the `VSAS` option if you plan to process the data with a SAS program that uses BY variable techniques. You must specify the SAS data set name using the `OUTSAS=` option.

B.13 WINDOWS Statement

A single WINDOWS statement may precede the EVTSTUDY and OLDSTUDY statements, and must precede the EXTRACT statement.

For a single event date event study, use WINDOWS to list up to 99 event windows for which cumulative or compounded abnormal returns and test statistics are to be reported on the output. The earliest and latest possible dates are determined by the value of the PRE and POST options respectively. If the WINDOWS statement is not present, Eventus automatically supplies three windows, including (−1,0), the part of the event period preceding date −1, and the part following date 0. A WINDOWS statement with no windows listed suppresses the window output.

With a two date event study (TWIN option on the EVENTUS statement), the usual windows are not valid. Instead, specify descriptors for the two event dates on the WINDOWS statement as shown above. These descriptors may be up to 14 characters long; they need not be valid SAS names, but may not contain blank spaces.

When used with the EXTRACT statement, WINDOWS specifies the event windows to cumulate and store for further analysis. In this situation, $(-1, 0)$ is not automatically included. The WINDOWS statement syntax in this usage is the same as for the type of event study (single date or TWIN) EXTRACT is processing.

Appendix C

How Eventus Finds the CRSP Stock Database

C.1 CRSPAccess Format

Eventus looks for a Windows environment variable, Unix environment variable or OpenVMS logical variable named `CRSP_DSTK` to automatically identify the location of the daily CRSPAccess database and the environment variable or logical `CRSP_MSTK` to identify the monthly CRSPAccess database. The CRSP document CRSP*Access Release Notes* describes how to set these environment variables or logicals on supported systems. However, if for some reason the appropriate environment variable or logical is not set, it is possible to manually tell Eventus where the database is located. To do so, use a `filename crspdb` statement to point to the aggregate storage location (folder or directory on most systems) containing the data, and also specify the option `DBFNSTMT` on the `EVENTUS` statement, for example:

```
filename crspdb 'c:\da199912';
eventus dbfnstmt;
etc.
```

To use a different data frequency in the estimation period of an event study from that used in the event period (in conjunction with the `EVENTUS` statement option `ESTINTER`) when the `DBFNSTMT` option is in effect, the fileref `mcrspdb` must point to the aggregate storage location containing the CRSPAccess database from which Eventus is to read the estimation period returns.

If the EVENTUS statement option EXCESS is specified, the SAS filerefs statidx1 and statidx2 must point to the daily files dsbo.dat and dsbc.dat, respectively, or to the corresponding monthly files msbo.dat and msbc.dat. The files are available from the ascii folder of the Indices CD-ROM.

C.2 SFA Format

To work with the CRSP SFA database format, the SAS filerefs crspstok should point to the standard stock data file and the fileref crspindx should point to the stock-associated calendar/indices file. If the Supplemental Nasdaq stock data file is to be read (by the Eventus PRICES or VOLUME statements), the fileref crspnms should point to it.

The same filerefs are used for daily and monthly data. Therefore, to change from daily to monthly usage, the user needs to specify the EVENTUS statement option MONTHLY, and also change the data files to which the above filerefs point.

To use a different data frequency in the estimation period of an event study from that used in the event period (in conjunction with the EVENTUS statement option ESTINTER), the filerefs crspes and estindx must point to the CRSP SFA standard stock data file and stock-associated calendar/indices file from which Eventus is to read the estimation period returns.

C.3 Size index files used with both CRSPAccess and SFA format databases

To use the CRSP Indices/Decile product (in conjunction with the REQUEST statement option SIZEINDX=CRSP), the fileref sizeindx should point to the CRSP Stock File Decile Indices file.

To use a size index file built by a program included for the purpose in the Eventus installation kit, (in conjunction with the REQUEST statement option SIZEINDX), the fileref sizeindx must point to it.

C.4 CRSP data permanently stored in SAS data sets

Eventus can use CRSP data stored in permanent SAS data sets, provided that a given SAS data library (usually a single folder or directory) contains only daily data or only monthly data, not a mixture of the two. When the `EVENTUS` statement option `SASCRSP` is set by default or specified at run time, Eventus expects the SAS data library to be used for the current run to be associated with the libref `CRSP`. For example,

```
libname crsp 'H:\SASCRSP\Daily';
eventus;
etc.
```

indicates that the SAS data sets containing the CRSP data for the current run are in the folder `SASCRSP\Daily` on drive `H` of a Windows computer or network. The libref that Eventus looks for can be changed using `LIBNAME` option on the `EVENTUS` statement.

To use a different data frequency in the estimation period of an event study from that used in the event period (in conjunction with the `EVENTUS` statement option `ESTINTER`), the libref specified in the `EVENTUS` statement option `ELIBNAME` must point to the SAS data library containing the CRSP data to be used for the estimation period. For example,

```
libname crsp '\home\crsp\sasdata\daily';
libname estcrsp '\home\crsp\sasdata\monthly';
eventus estinter=month elibname=estcrsp;
etc.
```

could be used to run a daily data event study with parameters estimated from monthly data, on a Unix system where the daily and monthly CRSP data are stored in SAS data sets in the directories `\home\crsp\sasdata\daily` and `\home\crsp\sasdata\monthly` respectively.

To use a size index file (in conjunction with the `REQUEST` statement option `SIZEINDX`), the `EVENTUS` statement option `SIZEIND` must point to the SAS data set containing the size index data. For example,

```
libname crsp '\home\crsp\sasdata\daily';
eventus sizeind=crsp.sizeindx;
request sizeindx;
etc.
```

indicates that the size index data set is in the same CRSP data library (directory \home\crsp\sasdata\daily in this case) as the main daily data and has the member name sizeindx.

References

Arshadi, Nasser, and Thomas H. Eyssell. "Regulatory Deterrence and Registered Insider Trading: The Case of Tender Offers," *Financial Management,* 1991, 20(2), 30–39.

Bhagat, Sanjai, M. Wayne Marr, and G. Rodney Thompson. "The Rule 415 Experiment: Equity Markets," *Journal of Finance,* 1985, 40(5), 1385–1401.

Binder, John J. "The Event Study Methodology Since 1969," *Review of Quantitative Finance and Accounting,* 1998, 11(2), 111–137.

Boehmer, Ekkehart, Jim Musumeci, and Annette B. Poulsen. "Event-Study Methodology under Conditions of Event-Induced Variance," *Journal of Financial Economics,* 1991, 30(2), 253–272.

Bollerslev, Tim, Ray Y. Chou, and Kenneth F. Kroner. "ARCH Modeling in Finance: A Review of the Theory and Empirical Evidence," *Journal of Econometrics,* 1992, 52(1/2), 5–60.

Brickley, James A., Frederick H. Dark, and Michael S. Weisbach. "The Economic Effects of Franchise Termination Laws," *Journal of Law and Economics,* 1991, 34(1), 101–132.

Brockett, Patrick L., Hwei-Mei Chen, and James R. Garven. "A new stochastically flexible event methodology with application to Proposition 103," *Insurance Mathematics and Economics,* 1999, 25(2), 197–217.

Brown, Stephen J., and Jerold B. Warner. "Measuring Security Price Performance," *Journal of Financial Economics,* 1980, 8(3), 205–258.

Brown, Stephen J., and Jerold B. Warner. "Using Daily Stock Returns: The Case Of Event Studies," *Journal of Financial Economics,* 1985, 14(1), 3–31.

Cantrell, Steve, Michael T. Maloney, and Mark Mitchell. "On Estimating the Variance of Abnormal Stock Market Performance," working paper, 1989, Clemson University.

Chandra, Ramesh, and Bala V. Balachandran. "A Synthesis of Alternative Testing Procedures for Event Studies," *Contemporary Accounting Research,* 1990, 6(2), 611–640.

Chandra, Ramesh, Shane Moriarity, and G. Lee Willinger. "A Reexamination of the Power of Alternative Return-Generating Models and the Effect of Accounting for Cross-Sectional Dependencies in Event Studies," *Journal of Accounting Research,* 1990, 28(2), 398–408.

Chen, Haiyang, Michael Y. Hu, and Joseph C.P. Shieh. "The Wealth Effect of International Joint Ventures: The Case of U.S. Investment in China," *Financial Management,* 1991, 20(4), 31–41.

Corhay, A. and A. Tourani Rad. "Conditional Heteroscedasticity Adjusted Market Model and an Event Study," *Quarterly Review of Economics and Finance,* 1996, 36(4), 529–538.

Collins, Daniel W. and Warren T. Dent. "A Comparison of Alternative Methodologies used in Capital Market Research," *Journal of Accounting Research,* 1984, 22(1), 48–84.

Corrado, Charles J. "A Nonparametric Test for Abnormal Security-Price Performance in Event Studies," *Journal of Financial Economics,* 1989, 23(2), 385–396.

Corrado, Charles J., and Terry L. Zivney, "The Specification and Power of the Sign Test in Event Study Hypothesis Tests Using Daily Stock Returns," *Journal of Financial and Quantitative Analysis,* 1992, 27(3), 465–478.

Cowan, Arnold R. "Nonparametric Event Study Tests," *Review of Quantitative Finance and Accounting,* 1992, 2(4), 343–358.

Cowan, Arnold R. "Tests for Cumulative Abnormal Returns over Long Periods: Simulation Evidence," *International Review of Financial Analysis,* 1993, 2(1), 51–68.

Cowan, Arnold R., Nandkumar Nayar, and Ajai K. Singh. "Stock Returns Before and After Calls of Convertible Bonds," *Journal of Financial and Quantitative Analysis,* 1990, 25(4), 549–554.

Cowan, Arnold R., and Anne M.A. Sergeant. "Trading Frequency and Event Study Test Specification," *Journal of Banking and Finance,* 1996, (10), 1731–1757.

Dopuch, Nicholas, Robert W. Holthausen, and Richard W. Leftwich. "Abnormal Stock Returns Associated with Media Disclosures of 'Subject to' Qualified Audit Opinions," *Journal of Accounting and Economics,* 1986, 8(2), 93–117.

Draper, Norman R., and Harry Smith. *Applied Regression Analysis,* Second Edition, (John Wiley & Sons, New York, 1981.)

Giaccotto, Carmelo, and James M. Sfiridis. "Hypothesis Testing in Event Studies: The Case of Variance Changes," *Journal of Economics and Business,* 1996, 48(4), 349–370.

Haw, In-Mu, Victor S. Pastena, and Steven B. Lilien. "Market Manifestation of Nonpublic Information Prior to Mergers: The Effect of Capital Structure," *Accounting Review,* 1990, 65(2), 432–451.

Karafiath, Imre."Using Dummy Variables In The Event Methodology," *Financial Review,* 1988, 23(3), 351–358.

Karafiath, Imre, and David E. Spencer. "Statistical Inference in Multiperiod Event Studies," *Review of Quantitative Finance and Accounting,* 1991, 1(4), 353–371.

Kramer, Lisa. "Alternative Methods for Robust Analysis in Event Study Applications," working paper, Simon Fraser University, 2000. Available from the World Wide Web, http://www.sfu.ca/ kramer/Papers/event.pdf.

Lee, D. Scott. "Management Buyout Proposals and Inside Information," *Journal of Finance,* 1992, 47(3), 1061–1079.

Linn, Scott C., and John J. McConnell. "An Empirical Investigation of the Impact of 'Antitakeover' Amendments on Common Stock Prices," *Journal of Financial Economics*, 1983, 11(1), 361–399.

MacKinlay, A. Craig. "Event Studies in Economics and Finance," *Journal of Economic Literature,* 1997, 35(1), 13–39.

Mais, Eric L., William T. Moore, and Ronald C. Rogers. "A Re-Examination of Shareholder Wealth Effects of Calls of Convertible Preferred Stock," *Journal of Finance,* 1989, 44(5), 1401–1410.

Malatesta, Paul H. "Measuring Abnormal Performance: The Event Parameter Approach Using Joint Generalized Least Squares," *Journal of Financial and Quantitative Analysis,* 1986, 21(1), 27–38.

Mann, Steven V., and Neil W. Sicherman. "The Agency Costs of Free Cash Flow: Acquisition Activity and Equity Issues," *Journal of Business,* 1991, 64(2), 213–227.

Mikkelson, Wayne H., and M. Megan Partch. "Valuation Effects of Security Offerings and the Issuance Process," *Journal of Financial Economics,* 1986, 15(1/2), 31–60.

Mikkelson, Wayne H., and M. Megan Partch. "Withdrawn Security Offerings," *Journal of Financial and Quantitative Analysis,* 1988, 23(2), 119–134; "Errata", 1988, 23(4), 487.

Neter, John, William Wasserman, and Michael H. Kutner. *Applied Linear Regression Models* (Richard D. Irwin, Inc., Homewood, Illinois, 1983.)

O'Hara, Maureen, and Wayne Shaw. "Deposit Insurance and Wealth Effects: The Value of Being 'Too Big to Fail'," *Journal of Finance,* 1990, 45(5), 1587–1600.

Patell, James M. "Corporate Forecasts of Earnings Per Share and Stock Price Behavior: Empirical Tests," *Journal of Accounting Research*, 1976, 14(2), 246–274.

Peterson, Pamela P. "Event Studies: A Review of Issues and Methodology," *Quarterly Journal of Business and Economics,* 1989, 28(3), 36–66.

Pilotte, Eugene. "Growth Opportunities and the Stock Price Response to New Financing," *Journal of Business,* 1992, 65(3), 371–394.

Prabhala, N. R. "Conditional Methods in Event Studies and an Equilibrium Justification for Standard Event-Study Procedures," *Review of Financial Studies,* 1997, 10(1), 1–38.

Rozeff, Michael S. and Mir A. Zaman. "Market Efficiency And Insider Trading: New Evidence," *Journal of Business,* 1988, 61(1), 25–44.

Sanders, Ralph W., Jr. and Russell P. Robins. "Discriminating between Wealth and Information Effects in Event Studies in Accounting and Finance Research," *Review of Quantitative Finance and Accounting,* 1991, 1(3), 307–329.

Sanger, Gary C. and James D. Peterson. "An Empirical Analysis Of Common Stock Delistings," *Journal of Financial and Quantitative Analysis,* 1990, 25(2), 261–272.

Schipper, Katherine and Abbie Smith. "Effects of Recontracting on Shareholder Wealth: The Case of Voluntary Spin-Offs," *Journal of Financial Economics*, 1983, 12(4), 437–468.

Scholes, Myron and Joseph T. Williams. "Estimating Betas from Nonsynchronous Data," *Journal of Financial Economics,* 1977, 5(3), 309–327.

Singh, Ajai K., Arnold R. Cowan, and Nandkumar Nayar. "Underwritten Calls of Convertible Bonds," *Journal of Financial Economics,* 1991, 29(1), 173–196.

Sprent, Peter. *Applied Nonparametric Statistical Methods* (Chapman and Hall, London, 1989.)

Swary, Itzhak. "Stock Market Reaction to Regulatory Action in the Continental Illinois Crisis," *Journal of Business,* 1986, 59(3), 451–474.

Sweeney, Richard J., "Levels of Significance in Event Studies," *Review of Quantitative Finance and Accounting,* 1991, 1(4), 373–382.

Index

Ansley, Craig 108
arbitrage portfolio 19, 151
Arshadi, Nasser 19
ask and bid prices 90
AUTODATE option 20, 86, 96, 145
    with weekly returns 16
Balachandran, Bala V. 32, 55
beta
    GARCH 26
    Scholes-Williams 26, 104
Bhagat, Sanjai 26
bid and ask prices 90
binary ASCII output files 87, 141, 152, 155
Binder, John J. 1
binomial test 112
Boehmer, Ekkehart 28, 52, 109, 111, 135, 140
Bollerslev, Tim 26, 51
BOOT option 30, 130
bootstrapped tests 130, 139
    1- or 2-tailed 130, 139
bootstrapping 30
BOTH option 25, 50
Brickley, James A. 111
Brockett, Patrick L. 26, 50
Brown, Stephen J. 111
buy-and-hold returns 30, 53, 104
    with EXTRACT statement 67
BUYHOLD option 30, 53
Cantrell, Steve 108
cash distributions 90, 141
Chandra, Ramesh 32, 55
Chen, Haiyang 112
Chen, Hwei-Mei 26, 50
Chou, Ray 26, 51
Collins, Daniel W. 29, 52
Collins-Dent test 29, 52
comparison period returns 25, 50, 66, 105
compounded returns 30, 53, 104
continuously compounded returns 22
copyright notice 2
Corhay, Albert 26, 50
Corrado, Charles J. 29, 53, 112
Cowan, Arnold R. 26, 29, 50, 52, 53, 108, 111, 112
CP option 25, 50
cross-sectional analysis 63
cross-sectional regression 65
    example 69
cross-sectional standard deviation 111
cross-sectional standard error 111
cross-sectional test 28, 51
    formula 111
CRSP database
    CRSPAccess 159
        environment variables 159
    excess returns 15, 26, 84, 88, 127
    how Eventus finds 159

Indices 21
monthly 128
Nasdaq-only and NYSE-AMEX only, together 126
old Nasdaq 128, 156
pre-1995 129
reading multiple databases in one run 129
SAS data sets 161
SFA 160
Supplemental Nasdaq 89, 91, 142, 157
CRSP share type code 89
CRSPDAY 148
CRSPDAY1 148
CRSPDAY2 148
cumulative returns 30, 53
CUSIP option 145
CUSIPERM option 17, 145
CUSIPERM statement 121
options
COLUMN 101, 121
EXTFILE 101, 121
REQFILE 121
Dark, Frederick H. 111
DATECONV statement
options
AUTODATE 96, 122
CUSIP 122
CUSIPERM 95, 122
DATEFMT 96, 122
EXTFILE 98, 122
GROUP 95, 123
GRWEIGHT 95, 123
ID 123
IDFMT 123
INSAS2 124
NDAYS 97
OUTDTFMT 97, 124
OUTSAS 98, 124
REQFILE 125
REQUEST2 125
SHIFT*n* 97, 125
SHORT 96
SORTBYID 97
UPCUSIP 126
dates
converting to CRSP calendar 93
CRSP day numbers 3, 20, 96
format of 20, 46, 73, 85, 96, 97, 122, 145
monthly data 93
multiperiod 26
non-trading 20, 86, 96, 122, 145
old Nasdaq files 145
paired 37, 59
shifting 150
weekly data 93
Dent, Warren T. 29, 52
DISTRIB option 90
dividends 90, 141
Dopuch, Nicholas 111
dummy variables 71
EGARCH errors 26
estimation 103
estimation period 22, 46, 74, 146
different return interval from event period 15, 115
event parameter approach 74
firm-specific 146
holding fixed over multiple event studies 146
length 22, 47, 74, 147
missing returns in 23, 75, 118, 149
split pre- and post-event 22, 74, 146

ESTLEN option 22, 47, 74
event dates
converting to CRSP calendar 93
CRSP day numbers 3, 20, 96
non-trading 20, 86, 96, 122, 145
paired 37
shifting 150
event parameter approach 71, 133
EVENTDA1 148
EVENTDA2 148
EVENTDAT 148
events
CAARs between two 37
EventStream software 43
EVENTUS statement
options
ACCESS97 126
ANNUAL 126
CHAR 126
CHAR4 126
DBFNSTMT 126
DUAL 126
ELIBNAME 16, 127
ESTINTER 15, 115, 127
EXCESS 15, 40, 84, 127
FBIN 127
GETDATA 83, 127
HOSTBIN 127
IDXLEAD 127
LIBNAME 128
MONTHLY 13, 83, 93, 128
NASDAQ 128
NLIBNAME 90, 128
NONCRSP 45, 128
PAGE 16, 45, 129
PRE95 129
QUARTER 129
REQFILES 129
SASCRSP 129
SIZEIND 129
TWIN 37, 40, 59, 97, 124, 129, 158
TWIN with DATECONV 93
VAXFMT 130
WEEKLY 93, 130
EVTSTUDY statement 5, 23, 25, 41, 48, 49, 50, 75, 130, 139
and EXTRACT 65
and OLDSTUDY 40
options
ALLDAYS 28, 51, 130
BOOT 30, 130
BOTH 25, 50, 136
BTAIL 130, 139
BUYHOLD 30, 53, 130
CDCSI 29, 52, 130
CP 25, 50, 131
CSECTERR 28, 51, 131
DETAIL 25, 49, 131
EGARCH 131
EGLS 29, 52, 131
FACTORS 76, 131
GARCH 131
INSAS 131
ITSUR 76, 132
JACKNIFE 29, 53, 132
MAXMISS 28, 132
MEDIAN 30, 53, 132
NOMAR 25, 50
NOMM 25, 50
NONAMES 24, 132
NOPLIST 24, 49, 132
NOPRINT 132
NOSTD 25, 50
NUMFM 133
OLSPARAM 76, 133

OUTSAS 31, 54, 133
OVERLAP 28, 51, 133
PACKAGE 31, 54, 133
POST 27, 51, 76, 134
PRE 27, 51, 76, 134
RANKTEST 29, 53, 134
RAW 25, 50
SERIAL 29, 52, 108, 134
SHRCODE 135
SIC 24, 135
SKIP 135
SPORT 26, 135
STDALL 25, 50, 135
STDCSECT 28, 52, 135
STDONLY 25, 50
SUR 76, 135
SW 26, 50, 104, 135
TAIL 31, 54, 135
TIMEUNIT 26, 136
VALUE 25, 50, 76, 136
REQUEST required before 145
EXTRACT statement
and PACKAGE 31, 54
example 67
options
BUYHOLD 67, 137, 138
CDCSI 137
EXTEND 66, 137
EXTFILE 137
ID 65, 137
IDFMT 137
INSAS 64
OUTSAS 137
SERIAL 67, 137
TYPE 66, 138
VALUE 138
VPREFIX 65, 138
WPREFIX 65
Eyssell, Thomas H. 19
FACTORS option 76
only allowed in NONCRSP mode 76
firm-by-firm results
exporting 63
for further analysis 63
printing 24
storing 31
GARCH errors 26
Garven, James 26, 50
generalized sign test 29, 37, 53, 112
Giaccotto, Carmelo 29, 53
group weight variable 147
grouping variable 17, 123, 147
Haw, In-Mu 105
holidays 20, 96
Holthausen, Robert W. 111
Hu, Michael Y. 112
industry codes 24, 88
interevent CAAR 37
iterated JGLS 76
iterated SUR 76, 132
jackknife test 29, 53, 132
joint generalized least squares 76, 132, 135
Karafiath, Imre 29, 52, 71, 108
Kramer, Lisa 30
Kroner, Kenneth 26, 51
Lee, D. Scott 108
Leftwich, Richard W. 111
Lilien, Steven B. 105
Linn, Scott C. 105
log-transformed returns 22
multiperiod 136
MacKinlay, A. Craig 1
Mais, Eric 108
Malatesta, Paul H. 71
Maloney, Michael T. 108

Mann, Steven V. 108
market adjusted returns 25, 49, 66, 110
market closed 20
market index
  excluding dividends 20, 149
  Nasdaq Composite 20, 151
  Standard & Poor's 20
market maker count 91, 156
market model
  abnormal return definition 103
  estimation 5
    GARCH 26
    Scholes-Williams 26
  multi-factor 76
  standardized abnormal return 106
Marr, M. Wayne 26
MAXMISS option 28
McConnell, John J. 105
mean adjusted returns 25, 50, 66, 105
MEDIAN option 30, 53
Mikkelson, Wayne H. 29, 52
MINESTN option 23, 75
missing returns
  code 118, 152
  estimation period 23, 75, 118, 147, 149
  event period 28, 66, 118
Mitchell, Mark 108
Moore, William T. 108
Moriarity, Shane 32, 55
multiperiod returns 26
Musumeci, Jim 28, 52, 109, 111, 135, 140
National Market System indicator 91, 156
Nayar, Nandkumar 108, 112
NDAYS option 86, 97
NODIVIDX option 20
  with RETURNS 87
NOMAR option 25, 41, 50
NOMM option 25, 41, 50
non-trading dates 20, 96, 122, 145
NONCRSP option 45
nonparametric tests 29, 53, 112
NOPLIST option 24, 49
NOSTD option 25, 41, 50
number of trades 91
O'Hara, Maureen 31, 32, 34, 54, 55, 56
OLDSTUDY statement 138
  options
    ALLDAYS 138
    BOOT 139
    BUYHOLD 139
    CDCSI 139
    CSECTERR 139
    ID 139
    JACKNIFE 139
    MEDIAN 140
    NOMAR 41
    NOMM 41
    NOPLIST 140
    NOSTD 41
    RANKTEST 140
    STDALL 41, 140
    STDCSECT 140
    STDONLY 41
    SW 140
    TAIL 140
output data file 92, 97, 98, 101
  CAARs 63
  CUSIPERM 101
  for cross-sectional analysis 63
  individual days 63
  PRICES 144

RETURNS 153
VOLUME 157
OUTSAS file contents
controlling 133
event studies
variable names 116
interpreting 116
OVERLAP option 28, 51
PACKAGE specifications 31, 54, 133
Partch, M. Megan 29, 52
Pastena, Victor S. 105
Patell, James 28, 52, 105
Peterson, James D. 112
Peterson, Pamela P. 1
Pilotte, Eugene 28, 51
POOL option 22, 74
POST option 27, 51, 76
Poulsen, Annette B. 28, 52, 109, 111, 135, 140
Prabhala, N. R. 1
PRE option 27, 51, 76
PRE95 option 129
precision-weighted returns 107
prices
bid/low and ask/high 89
Nasdaq bid and ask 89
split adjusted 143
PRICES statement 141
default output 144
options
BIDASK 89, 141
BINARY 87, 141
DISTRIB 90, 141
EXTFILE 92, 141
HSAS 88, 141
NMS 89, 142
NOCLOSE 90, 143
OUTSAS 92, 143
SHARES 90, 143
SHRCODE 89, 143
SIC 143
SPLITADJ 91, 143
TRADES 91, 144
VSAS 88, 144
REQUEST required before 145
printed output
orientation 16, 45, 129
suppressing 24, 48, 49, 75, 132
printing individual firm results 24
PROC REG 65, 69
Rad, A. Tourani 26, 50
rank test 29, 53, 134
RAW option 25, 50
raw returns 25, 50, 66, 83, 110
REG procedure (SAS) 65, 69
regression, cross-sectional 65
example 69
REQFILE option 121, 155
request file 3, 16, 19, 39, 45, 46, 61, 73, 85, 86, 96, 124, 146, 150, 155
event parameter approach 72
example 4, 33
for CUSIPERM 100
for DATECONV 94, 125
for PRICES 84
for RETURNS 84
for VOLUME 84
how to format 150
more than two 129
SAS data set as 147
second 149, 150
sorting 4
with CUSIPERM 121
with NDAYS 149
REQUEST statement

options
AUTODATE 16, 20, 84, 86, 145
AUTODATE=BACK 145
COMPOSIT 20, 87, 151
CUSIP 145
CUSIPERM 17, 145
DATEFMT 20, 46, 73, 86, 145
EST 22, 46, 74, 146
ESTLEN 22, 47, 74
GROUP 17, 147
GRWEIGHT 147
ID 17, 46, 73, 85
IDFMT 85
INSAS2 149
IX2Y 149
LOG 22, 136, 149
MINESTN 23, 75, 149
NAME 149
NDAYS 86, 149, 149
NODIVIDX 20, 87, 149
POOL 22, 74
REQFILE 150
REQFILE2 150
SHIFT*n* 150
SHORT 19, 151
SIZEINDX 21, 151
SP500 20, 87, 151
UPCUSIP 151
required before EVTSTUDY 145
required before PRICES 145
required before RETURNS 145
required before VOLUME 145
results, output for cross-sectional analysis 63
RETURNS statement 152
default output 153
options
BINARY 87, 152
BOTH 88, 153
EXTFILE 92, 152
HSAS 88
INDEX 88, 153
OUTSAS 92, 153
SHRCODE 89, 153
SIC 153
SPORT 88
VALUE 88, 153
VSAS 88, 152, 154
REQUEST required before 145
Robins, Russell P. 29, 52, 109
Rogers, Ronald C. 108
Rozeff, Michael S. 19
Sanders, Ralph W., Jr. 29, 52, 109
Sanger, Gary C. 112
SAS data sets
storage of CRSP data 128, 129, 161
variable names 116
Schipper, Katherine 105
Scholes, Myron M. 26, 50, 104
Scholes-Williams beta 26, 50, 66, 104
secondary prices 141
seemingly unrelated regressions 76, 132, 135
selling short 19, 151
Sergeant, Anne M. A. 26, 50
serial correlation
correction for 108
Sfiridis, James M. 29, 53
share type code 89, 135, 143, 153, 156
shares outstanding 90, 143, 156
Shaw, Wayne 31, 32, 34, 54, 55, 56
Shieh, Joseph C.P. 112
short position 19
SIC codes 24, 88, 135
SIC option 24
Sicherman, Neil W. 108

sign test 29, 53, 112
significance level 31, 54, 135
Singh, Ajai K. 108, 112
size-adjusted returns 21
Smith, Abbie 105
Spencer, David E. 29, 52, 108
split adjustment 156
Sprent, Peter 112
standard errors 28, 29, 51, 52
standardized abnormal return 52, 106
standardized cross-sectional test 28, 52
  EVTSTUDY 135
  OLDSTUDY 140
STDALL option 25, 50
STDONLY option 25, 50
stock splits 91
Supplemental Nasdaq File 89, 91, 142, 157
SUR 76, 135
SW option 26, 50
Swary, Itzhak 31
Sweeney, Richard J. 108
TAIL option 31, 54, 135
test statistic
  Boehmer-Musumeci-Poulsen 28, 52
  bootstrap 30
  Collins-Dent 29, 52
  cross-sectional method 111
  EGLS 29, 52
  generalized sign 29, 53, 112
  jackknife 29, 53, 113
  Patell 28, 52, 109
  portfolio 110
  rank 29, 53, 112
  serial dependence adjustment 29, 52
  standardized abnormal return 28, 52, 109
  standardized cross-sectional 28, 52
Thompson, G. Rodney 26
time series standard deviation 110
TIMEUNIT option 26
  versus WINDOWS 27
trademarks 2
TRADES option 91
trading status 91, 156
trading volume 155
TWIN option 37, 59
UPCUSIP option 151
UPCUSIP statement 154
  options
    COLUMN 154
    EXTFILE 155
    REQFILE 155
USERSTOK file 43
  using SAS data set 48, 131
    example 78
VALUE option 25, 50
value-weighted index 25, 50, 87, 88, 136
variable names 116
  WEIGHT 116
  CRSPDAY 148
  CRSPDAY1 148
  CRSPDAY2 148
  event study OUTSAS data set 117
  EVENTDA1 148
  EVENTDA2 148
  EVENTDAT 148
  for group weight 148
  in INSAS for non-CRSP 131
  in INSAS for request file 147
  in PRICES OUTSAS data set 144
  in RETURNS OUTSAS data set 154

in VOLUME OUTSAS data set 157
RESTYPE 118
with DATEFMT=CRSP 149
variance increase 28, 52, 111
VAXFMT option 130
VAXRB4. SAS informat 130
VOLUME statement 155
default output 157
options
BINARY 87, 155
EXTFILE 155
HSAS 88
NASDINFO 91, 156
OUTSAS 92, 156
SHARES 90, 92, 156
SHRCODE 89, 156
SIC 156
SPLITADJ 91, 156
TRADES 91, 157
VSAS 88, 157
REQUEST required before 145
Warner, Jerold B. 111
weight
firms within subgroups 147
negative 19
weighted least squares 65
example 69
with serial correlation adjustment 67
Weisbach, Michael S. 111
Williams, Joseph T. 26, 50, 104
Willinger, G. Lee 32, 55
window
variable length 37
WINDOWS statement 23, 39, 40, 48, 61, 158
event parameter approach 75
versus TIMEUNIT 27
with EVTSTUDY 25, 27, 49, 51
TWIN 39, 61, 158
with EXTRACT 64, 158
with OLDSTUDY 40
Zaman, Mir A. 19
Zivney, Terry L. 112